



DAGSTUHL REPORTS

Volume 12, Issue 11, November 2022

Principles of Contract Languages (Dagstuhl Seminar 22451) <i>Dilian Gurov, Reiner Hähnle, Marieke Huisman, Giles Reger, and Christian Lidström</i>	1
Computational Social Dynamics (Dagstuhl Seminar 22452) <i>Martin Hofer, Sigal Oren, Roger Wattenhofer, and Giovanna Varricchio</i>	28
Dynamic Graph Algorithms (Dagstuhl Seminar 22461) <i>Aaron Bernstein, Shiri Chechik, Sebastian Forster, Tsvi Kopelowitz, Yasamin Nazari, and Nicole Wein</i>	45
Set Visualization and Uncertainty (Dagstuhl Seminar 22462) <i>Susanne Bleisch, Steven Chaplick, Jan-Henrik Haunert, Eva Mayr, Marc van Kreveld, and Annika Bonerath</i>	66
Towards More Flexible and Automated Communication Networks (Dagstuhl Seminar 22471) <i>Artur Hecker, Stefan Schmid, Henning Schulzrinne, Lily Hügerich, Sándor Laki, and Iosif Salem</i>	96
Vertex Partitioning in Graphs: From Structure to Algorithms (Dagstuhl Seminar 22481) <i>Maria Chudnovsky, Misra, Neeldhara, Daniel Paulusma, Oliver Schaudt, and Akanksha Agrawal</i>	109
Counting and Sampling: Algorithms and Complexity (Dagstuhl Seminar 22482) <i>Holger Dell, Mark R. Jerrum, Haiko Müller, Konrad Anand, and Marcus Pappik</i>	124

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

May, 2023

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.

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
- Daniel Cremers
- Goetz Graefe
- Reiner Hähnle
- Barbara Hammer
- Lynda Hardman
- Oliver Kohlbacher
- Steve Kremer
- Rupak Majumdar
- Heiko Mantel
- 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>

Digital Object Identifier: 10.4230/DagRep.12.11.i

Principles of Contract Languages

Dilian Gurov^{*1}, Reiner Hähnle^{*2}, Marieke Huisman^{*3}, Giles Regeer^{*4},
and Christian Lidström^{†5}

- 1 KTH Royal Institute of Technology – Stockholm, SE. dilian@kth.se
- 2 TU Darmstadt, DE. haehnle@cs.tu-darmstadt.de
- 3 University of Twente – Enschede, NL. m.huisman@utwente.nl
- 4 University of Manchester, GB. giles.regeer@manchester.ac.uk
- 5 KTH Royal Institute of Technology – Stockholm, SE. clid@kth.se

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 22451 “Principles of Contract Languages”. At the seminar, participants discussed the fundamental aspects of software contracts. Topics included the format and expressiveness of contracts, their use cases in software development and analysis, and contract composition and decomposition.

Seminar November 6–11, 2022 – <http://www.dagstuhl.de/22451>

2012 ACM Subject Classification Theory of computation → Program specifications

Keywords and phrases software contracts, program specifications, software development, program analysis

Digital Object Identifier 10.4230/DagRep.12.11.1


1 Executive Summary

Dilian Gurov

Reiner Hähnle

Marieke Huisman

Giles Regeer

License  Creative Commons BY 4.0 International license
© Dilian Gurov, Reiner Hähnle, Marieke Huisman, and Giles Regeer

This report documents the program and the outcomes of Dagstuhl Seminar 22451 “Principles of Contract Languages”.

Formal, precise analysis of non-trivial software is a task that necessarily must be decomposed. The arguably most important composition principle in programming is the procedure (function, method, routine) call. For this reason, it is natural to decompose the analysis of a program along its call structure. Decomposition in this context means to replace a procedure call with a declarative description, possibly an approximation, of the call’s effect. In his seminal work on runtime verification in Eiffel, Bertrand Meyer suggested to use the metaphor of a contract between the user (caller) and implementor (callee) for such a description.

Contracts continue to be a central element in run-time (dynamic) analysis. In the last two decades they also became the dominant decomposition approach in deductive verification and are realized in all major software verification systems. More recently, software contracts are increasingly used in test case generation and model checking. Furthermore, programming

* Editor / Organizer

† Editorial Assistant / Collector



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

Principles of Contract Languages, *Dagstuhl Reports*, Vol. 12, Issue 11, pp. 1–27

Editors: Dilian Gurov, Reiner Hähnle, Marieke Huisman, Giles Regeer, and Christian Lidström



Dagstuhl Reports

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

languages such as “Racket” or “Dafny” were designed with a notion of contract. Contract-based specification languages are available for mainstream programming languages, notably JML for “Java” and ACSL/ACSL++ for “C”/“C++”.

However, there is considerable fragmentation concerning terminology, basic principles, expressivity, and usage of contracts in different research communities. Therefore, this Dagstuhl Seminar convened researchers working with contracts in static verification, runtime verification, as well as testing, with the goal of creating a unified view on software contracts.

The seminar participants discussed the following topics and questions:

1. *Sub-procedural* contracts: contracts for blocks, loops, suspension points, barriers
2. Combining *trace-based specifications* for *global* properties with *two-state* contracts for recursive procedures
3. Rethink *abstract* versus implementation-layer specifications
4. Contracts for *concurrent languages*
5. Contract *composition*
6. Unify contracts for deductive and runtime verification and testing
7. (Behavioral) *types* as lightweight contracts
8. Where do contracts come from? Contract *synthesis*
9. Contract *validation*, connection to *natural language*
10. Contracts for *refinement* (correctness by construction)
11. Contracts for relational properties
12. Debugging contracts
13. Domain-specific contract languages

2 Table of Contents

Executive Summary

<i>Dilian Gurov, Reiner Hähnle, Marieke Huisman, and Giles Reger</i>	1
--	---

Overview of Talks

Industrial Experience with Specification <i>David Cok</i>	5
Executable Contracts in Ada and SPARK <i>Claire Dross</i>	5
Loop Verification with Invariants and Contracts <i>Gidon Ernst</i>	5
Explaining Contract Violations with BeepBeep 3 <i>Sylvain Hallé</i>	6
Contracts in Model Checking <i>Paula Herber</i>	6
A Program Logic for Data Dependence Analysis <i>Asmae Heydari Tabar</i>	7
Formal Specification with Contracts: Past, Present, Future <i>Reiner Hähnle</i>	7
A Contract-based View on Functional Equivalence Checking <i>Marie-Christine Jakobs</i>	8
Towards Domain-Specific Contracts for Programs <i>Eduard Kamburjan</i>	8
Formal Verification of a JavaCard Virtual Machine for Common Criteria Certification <i>Nikolai Kosmatov</i>	9
From English Assume-Guarantee Contracts to Validated Temporal Logic Specifications <i>Kristin Yvonne Rozier</i>	9
Efficient First-Order Runtime Verification <i>Srdan Krstic</i>	10
Formal Specifications Investigated: A Classification and Analysis of Annotations for Deductive Verifiers <i>Sophie Lathouwers</i>	10
Contracts, abstractly <i>Christian Lidström</i>	11
Contracts in Static Verification <i>Rosemary Monahan</i>	11
On Automatic Contract Synthesis <i>Philipp Rümmer</i>	12
When You Hit the Frame <i>Thomas Santen</i>	13

Correctness-by-Construction: The CorC Ecosystem <i>Ina Schaefer</i>	13
VeyMont: Parallelising Verified Programs instead of Verifying Parallel Programs <i>Petra van den Bos</i>	13
Working groups	
Two-state versus Trace-based Contracts: Report from Breakout Group 1 <i>Dilian Gurov, Bernhard Beckert, Alessandro Cimatti, Paula Herber, Srdan Krstic, Martin Leucker, Christian Lidström, and Marco Scaletta</i>	14
Abstraction in Contracts: Report from Breakout Group 2 <i>Reiner Hähnle, Asmae Heydari Tabar, Eduard Kamburjan, Nikolai Kosmatov, Rosemary Monahan, Thomas Santen, Ina Schaefer, Julien Signoles, and Alexander J. Summers</i>	18
Specification Engineering: Report from Breakout Group 3 <i>Wolfgang Ahrendt, David Cok, Gidon Ernst, Sophie Lathouwers, Giles Reger, Kristin Yvonne Rozier, and Philipp Rümmer</i>	21
Interoperable Contracts: Report from Breakout Group 4 <i>Petra van den Bos, Claire Dross, Sylvain Hallé, Marieke Huisman, Marie-Christine Jakobs, and Mattias Ulbrich</i>	24
Participants	27

3 Overview of Talks

3.1 Industrial Experience with Specification

David Cok (Safer Software Consulting – Rochester, US)

License © Creative Commons BY 4.0 International license
© David Cok

Academic and workshop verification problems tend to focus on proving easily described and compact algorithms. When specifying and verifying legacy industrial code, however, such algorithmic verification is not the main challenge. Rather other aspects of DV are important: validating that formal specifications correspond to the real requirements, handling of frame conditions, both reads and modifies conditions, in large bodies of code; effective specification and verification combined with abstraction; and the ability to efficiently debug failed proofs.

3.2 Executable Contracts in Ada and SPARK

Claire Dross (AdaCore – Paris, FR)

License © Creative Commons BY 4.0 International license
© Claire Dross

This talk presents the contracts available in the Ada language. They have an executable semantics and can be checked both dynamically and statically through the deductive verification tool SPARK. The talk mostly focuses on the challenges that we are currently facing both for dynamic analysis (generation of test values which comply with the precondition, coverage of the postcondition...) and for deductive verification (support of non-executable constructs such as quantification or logical equality, contracts for pointer support and ownership...).

3.3 Loop Verification with Invariants and Contracts

Gidon Ernst (LMU München, DE)

License © Creative Commons BY 4.0 International license
© Gidon Ernst

Main reference Gidon Ernst: “Loop Verification with Invariants and Contracts”, in Proc. of the Verification, Model Checking, and Abstract Interpretation – 23rd International Conference, VMCAI 2022, Philadelphia, PA, USA, January 16-18, 2022, Proceedings, Lecture Notes in Computer Science, Vol. 13182, pp. 69–92, Springer, 2022.

URL https://doi.org/10.1007/978-3-030-94583-1_4

Invariants are the predominant approach to verify the correctness of loops. As an alternative, loop contracts, which make explicit the premise and conclusion of the underlying induction proof, can sometimes capture correctness conditions more naturally. But despite this advantage, the second approach receives little attention overall. In this talk we explore the fundamentals of loop contracts in an accessible way, demonstrate constructive translations from and to purely invariant-based proofs, characterize completeness of both approaches, and discuss their relative strengths and weaknesses on well-known verification problems, cf. [1]. Finally, we point out a few ideas on how loop contracts can perhaps lead to novel ways of automating correctness proofs of loopy programs.

References

- 1 Gidon Ernst. *Loop Verification with Invariants and Contracts*. In Proc. of VMCAI, Springer, LNCS, 2022.

3.4 Explaining Contract Violations with BeepBeep 3

Sylvain Hallé (University of Quebec at Chicoutimi, CA)

License © Creative Commons BY 4.0 International license
© Sylvain Hallé

Main reference Sylvain Hallé: “Explainable Queries over Event Logs”, in Proc. of the 24th IEEE International Enterprise Distributed Object Computing Conference, EDOC 2020, Eindhoven, The Netherlands, October 5-8, 2020, pp. 171–180, IEEE, 2020.

URL <https://doi.org/10.1109/EDOC49727.2020.00029>

It is one thing to discover that the conditions stipulated by a contract have been violated –but it is another thing to provide an explanation for this violation. And what is an explanation anyway? In this short talk, we explored the notion of explanation for specifications expressed on event streams, and showed how an existing event stream processing engine called BeepBeep has been retrofitted to automatically compute explanations for the violation of a specification.

3.5 Contracts in Model Checking

Paula Herber (Universität Münster, DE)

License © Creative Commons BY 4.0 International license
© Paula Herber

Joint work of Paula Herber, Timm Liebrez, Pauline Blohm

Main reference Paula Herber, Timm Liebrez: “Dependence Analysis and Automated Partitioning for Scalable Formal Analysis of SystemC Designs”, in Proc. of the 18th ACM/IEEE International Conference on Formal Methods and Models for System Design, MEMOCODE 2020, Jaipur, India, December 2-4, 2020, pp. 1–6, IEEE, 2020.

URL <https://doi.org/10.1109/MEMOCODE51338.2020.9314998>

In this talk, I have started with a brief history of model checking, introduced the general idea, and sketched some major achievements that have significantly increased the feasibility of model checking over the years. Then, I have presented two examples from our own work where we have applied model checking to embedded systems: An approach where we proposed automated partitioning of hardware/software co-designs and then model checked the partitions, and another approach where we proposed to use contracts for the safe integration of reinforcement learning into embedded control systems. I have discussed some ideas we had for modular verification that actually enabled us to verify complex systems, but I also demonstrated where we failed and where I think introducing a notion of contracts into the verification process would be beneficial, if good abstractions and constructs to express these abstraction were available.

References

- 1 Edmund M., Clarke Jr., Orna Grumberg, Daniel Kroening, Doron Peled and Helmut Veith. *Model Checking. Second Edition*. MIT 2018.
- 2 Paula Herber, Timm Liebrez. *Dependence Analysis and Automated Partitioning for Scalable Formal Analysis of SystemC Designs*. In Proceedings of the ACM and IEEE International Conference on Formal Methods and Models for Co-Design (MEMOCODE), 2020.

3.6 A Program Logic for Data Dependence Analysis

Asmae Heydari Tabar (TU Darmstadt, DE)

License © Creative Commons BY 4.0 International license
© Asmae Heydari Tabar

Joint work of Asmae Heydari Tabar, Richard Bubel, Reiner Hähnle

Main reference Asmae Heydari Tabar, Richard Bubel, Reiner Hähnle: “Automatic Loop Invariant Generation for Data Dependence Analysis”, in Proc. of the 10th IEEE/ACM International Conference on Formal Methods in Software Engineering, FormaliSE@ICSE 2022, Pittsburgh, PA, USA, May 22-23, 2022, pp. 34–45, ACM, 2022.

URL <https://doi.org/10.1145/3524482.3527649>

Parallelization of programs relies on sound and precise analysis of data dependences in the code, specifically, when dealing with loops. State-of-art tools are based on dynamic profiling and static analysis. They tend to over- and, occasionally, to under-approximate dependences. The former misses parallelization opportunities, the latter can change the behavior of the parallelized program. We have developed a sound and highly precise approach to generate data dependences based on deductive verification. The central technique is to infer a specific form of loop invariant tailored to express dependences. To achieve full automation, we adapt predicate abstraction in a suitable manner. To retain as much precision as possible, we generalized logic-based symbolic execution to compute abstract dependence predicates. We implemented our approach for Java on top of a deductive verification tool. The evaluation shows that our approach can generate highly precise data dependences for representative code taken from HPC applications.

3.7 Formal Specification with Contracts: Past, Present, Future

Reiner Hähnle (TU Darmstadt, DE)

License © Creative Commons BY 4.0 International license
© Reiner Hähnle

Contract-based formal specification languages are popular in deductive verification, runtime verification and, increasingly, in model checking and test case generation. We outline the building blocks of contract languages and trace some historical developments, as well as current usage. Finally, we suggest some research challenges:

1. *Sub-procedural* contracts
 - blocks, loops, suspension points, barriers
2. Combine *trace-based specification* for *global* properties with *lossless* contracts for recursive procedures
3. Rethink *abstract* vs. implementation-layer specs
4. Contracts for *concurrent languages*
5. Contract *composition*
6. Unify contracts for deductive & runtime verification and testing
7. (Behavioral) *types* as lightweight contracts
8. Where do contracts come from? Contract *synthesis*
9. Contract *validation*, connect to *natural language*
10. Contracts for *refinement* (correctness by construction)
11. Contracts for relational properties
12. Debugging contracts
13. Domain-specific contract languages

3.8 A Contract-based View on Functional Equivalence Checking

Marie-Christine Jakobs (TU Darmstadt, DE)

License © Creative Commons BY 4.0 International license
© Marie-Christine Jakobs

Main reference Marie-Christine Jakobs: “PEQCHECK: Localized and Context-aware Checking of Functional Equivalence”, in Proc. of the 9th IEEE/ACM International Conference on Formal Methods in Software Engineering, FormaliSE@ICSE 2021, Madrid, Spain, May 17-21, 2021, pp. 130–140, IEEE, 2021.

URL <https://doi.org/10.1109/FormaliSE52586.2021.00019>

So far, I have used software verifiers to check functional equivalence of two programs, the original and refactored one. To this end, I split the check for functional equivalence into one check for each refactored code segment [1] (code segments are identified by the user). Each check of a refactored code segment inspects the functional equivalence of original and refactored code segment. Therefore, I encode the check of each refactored code segment into a separate verification task. In this talk, I will look into an alternative to describe the verification task, namely contracts. This may allow us to use other tools, e.g., deductive verifiers to check the equivalence of the two code segments.

References

- 1 M.-C. Jakobs. PEQcheck: Localized and context-aware checking of functional equivalence. In *Proc. FormaliSE*, pages 130–140. IEEE, 2021.

3.9 Towards Domain-Specific Contracts for Programs

Eduard Kamburjan (University of Oslo, NO)

License © Creative Commons BY 4.0 International license
© Eduard Kamburjan

Semantically Lifted programs [1] enable programs to explicitly refer to their own program state as a *knowledge graph*, and add information about the application domain through the integration of ontologies. This talk discussed the challenges and chances for contracts in this setting: using ontologies, we can formulate contracts using domain knowledge *together* with the domain expert, yet stay in a fully formal setting thanks to the rich theories developed the semantic web. However, the questions how exactly to (a) connect with these theories, and (b) how to use languages for data access, such as SPARQL, in a contract language beyond simple typing guarantees [2] remain open.

References

- 1 Kamburjan, E., Klungre, V., Schlatte, R., Johnsen, E. & Giese, M. Programming and Debugging with Semantically Lifted States. *ESWC*. (2021)
- 2 Kamburjan, E. & Kostylev, E. Type Checking Semantically Lifted Programs via Query Containment under Entailment Regimes. *Description Logics*. (2021)

3.10 Formal Verification of a JavaCard Virtual Machine for Common Criteria Certification

Nikolai Kosmatov (Thales Research & Technology – Palaiseau, FR)

License © Creative Commons BY 4.0 International license
© Nikolai Kosmatov

Joint work of Adel Djoudi, Martin Hána, Nikolai Kosmatov

Main reference Adel Djoudi, Martin Hána, Nikolai Kosmatov: “Formal Verification of a JavaCard Virtual Machine with Frama-C”, in Proc. of the Formal Methods – 24th International Symposium, FM 2021, Virtual Event, November 20-26, 2021, Proceedings, Lecture Notes in Computer Science, Vol. 13047, pp. 427–444, Springer, 2021.

URL https://doi.org/10.1007/978-3-030-90870-6_23

Formal verification of real-life industrial software remains a challenging task. It provides strong guarantees of correctness, which are particularly important for security-critical products, such as smart cards. Security of a smart card strongly relies on the requirement that the underlying JavaCard virtual machine ensures necessary isolation properties. This talk presents a recent formal verification of a JavaCard Virtual Machine implementation performed by Thales for Common Criteria certification using the Frama-C verification toolset. The certification was successful: an EAL6 certificate was delivered in October 2021 and an EAL7 certificate (the highest one) was delivered in October 2022. This is the first verification project for such a large-scale industrial smart card product where deductive verification is applied on the real-life C code. The target properties include common security properties such as integrity and confidentiality. The implementation contains over 7,000 lines of C code. After a formal specification in the ACSL specification language, over 50,000 verification conditions were generated and successfully proved. We give an overview of the project and proof results, and focus on the mataproperty based approach that was a key solution for a successful specification and verification of security properties. This is a joint work with Adel Djoudi and Martin Hána.

3.11 From English Assume-Guarantee Contracts to Validated Temporal Logic Specifications

Kristin Yvonne Rozier

License © Creative Commons BY 4.0 International license
© Kristin Yvonne Rozier

Major safety-critical systems, like the NASA Lunar Gateway Vehicle System Manager, utilize structured English requirements (called contracts) with strict requirements for validation and traceability from early design time through system runtime. We overview the successes and challenges of this approach, highlight recent work in automated temporal logic contract validation, and point out the most immediate needs for future research.

3.12 Efficient First-Order Runtime Verification

Srdan Krstic (ETH Zürich, CH)

License © Creative Commons BY 4.0 International license
© Srdan Krstic

Joint work of Martin Raszyk, David A. Basin, Srdan Krstic, Dmitriy Traytel

Main reference Martin Raszyk, David A. Basin, Srdan Krstic, Dmitriy Traytel: “Practical Relational Calculus Query Evaluation”, in Proc. of the 25th International Conference on Database Theory, ICDT 2022, March 29 to April 1, 2022, Edinburgh, UK (Virtual Conference), LIPIcs, Vol. 220, pp. 11:1–11:21, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

URL <https://doi.org/10.4230/LIPIcs.ICDT.2022.11>

Existing arbitrary first-order (FO) formula evaluation approaches are less efficient than the established algorithms based on finite tables which are used in existing database management systems. However, latter cannot handle arbitrary FO formulas, but rather those that have a particular syntactic structure, e.g., isomorphic to relational algebra expressions.

I will showcase a new translation from arbitrary FO formula into two relational algebra expressions for which the finiteness of the formula’s evaluation result is guaranteed. Assuming an infinite domain, the two expressions have the following meaning: The first is closed and characterizes the original formula’s relative safety, i.e., whether given a fixed database, the original formula evaluates to a finite relation. The second expression is equivalent to the original formula, if it is relatively safe. The translation improves the time complexity over existing approaches, which we also empirically confirm in both realistic and synthetic experiments.

3.13 Formal Specifications Investigated: A Classification and Analysis of Annotations for Deductive Verifiers

Sophie Lathouwers (University of Twente – Enschede, NL)

License © Creative Commons BY 4.0 International license
© Sophie Lathouwers

Joint work of Sophie Lathouwers, Marieke Huisman

Main reference Sophie Lathouwers, Marieke Huisman: “Formal Specifications Investigated: A Classification and Analysis of Annotations for Deductive Verifiers”, in Proc. of the 10th IEEE/ACM International Conference on Formal Methods in Software Engineering, FormaliSE@ICSE 2022, Pittsburgh, PA, USA, May 22-23, 2022, pp. 69–79, ACM, 2022.

URL <https://doi.org/10.1145/3524482.3527652>

This talk investigates what annotations are actually needed for deductive verification, and provides a taxonomy to categorise these annotations. In particular, we identify several top-level categories, which are further divided into subcategories of annotations. This taxonomy is then used as a basis to investigate how often particular annotation categories occur, by inspecting over 10k lines of annotated programs. To determine whether the results are in line with expectations, we have interviewed several experts on deductive verification. Moreover, we show how the results can be used to evaluate the effectiveness of annotation generators. The knowledge from this analysis provides a gateway to guide further research in improving the efficiency of deductive verification, e.g.: it can serve as a guideline on what categories of annotations should be generated automatically, to evaluate the power of existing annotation generation techniques, and to improve the teaching of deductive verification.

3.14 Contracts, abstractly

Christian Lidström (KTH Royal Institute of Technology – Stockholm, SE)

License © Creative Commons BY 4.0 International license
© Christian Lidström

In our view, the primary purpose of contracts is to split responsibilities, and assign them to different parts of a system. To this end, we propose an abstract view of contracts as the formal foundation. Treating contracts as mathematical objects enable elegant formalisation of their properties, while also allowing artefacts of existing formalisms and tools to be cast in the theory.

We discuss our work on a contract theory, based in this principle. The contract theory is aimed at procedural, embedded software, and adheres to established desired properties in system design. It is defined only at the semantic level, in a denotational style. It is also parametrised on the semantic domain, enabling the instantiation of different semantics for particular uses. Previously, we have verified industrial embedded software deductively, with the goal of ensuring functional correctness of low-level system components. Recently we are also interested in high-level system properties, typically of a temporal nature. Because of the abstract view taken of contracts, both of these notions fit naturally into our framework.

3.15 Contracts in Static Verification

Rosemary Monahan (National University of Ireland – Maynooth, IE)

License © Creative Commons BY 4.0 International license
© Rosemary Monahan

Joint work of Marie Farrell, Conor Reynolds, Rosemary Monahan

Main reference Marie Farrell, Conor Reynolds, Rosemary Monahan: “Using dafny to solve the VerifyThis 2021 challenges”, in Proc. of the FTfJP 2021: 23rd ACM International Workshop on Formal Techniques for Java-like Programs, Virtual Event, Denmark, 13 July 2021, pp. 32–38, ACM, 2021.

URL <https://doi.org/10.1145/3464971.3468422>

We overview the structure of contracts languages used for specifying code for static verification. We report on our experience of using the Dafny Language in teaching software verification at both undergraduate and postgraduate level at Maynooth University; using Dafny to complete verification challenges in our research groups; and our experience of similar tools which have participated in the VerifyThis verification competition series over the past decade.

Contracts in static verification focus on modular specification of code. We present a range of examples and exercises with specification contracts consisting of pre-conditions, post-conditions and frame conditions, indicated by the **requires**, **ensures** and **modifies** clauses respectively. We discuss under-specification and over-specification in contracts, directing discussion to what is meant by contract verification. We note that verification proofs often require more information than that provided in the method contracts. This includes information expressed as assertions (**assert**), loop and class invariants (**invariant**), termination metrics (**decreases**) and specification-only **ghost** code, as well as using built-in data types (**set**, **sequence**, **multiset** etc.) which provide abstraction and support for verification. We discuss their role in both the specification contracts for the code, and in the contracts which verification tools use during the proof of correctness. Finally, we summarise how we view contract’s in static verification and present future challenges for contract languages and static verification tools as the way that we build and verify software evolves.

References

- 1 M Farrell, C Reynolds, and R Monahan. *Using Dafny to Solve the VerifyThis 2021 Challenges* Proceedings of the 23rd ACM International Workshop on Formal Techniques for Java-like Programs, Association for Computing Machinery, 2021
- 2 M Farrell, M Bradbury, M Fisher, L Dennis, C Dixon, H Yuan and C Maple. *Using Threat Analysis Techniques to Guide Formal Verification: A Case Study of Cooperative Awareness Messages* International Conference on Software Engineering and Formal Methods, Springer, 2019
- 3 M Farrell, N Mavrikis, C Dixon and Y Gao. *Formal Verification of an Autonomous Grasping Algorithm* International Symposium on Artificial Intelligence, Robotics and Automation in Space, 2020
- 4 K Rustan M Leino. *Dafny: An automatic program verifier for functional correctness*. International Conference on Logic for Programming Artificial Intelligence and Reasoning, Springer, 2010
- 5 M Huisman, R Monahan, P Müller, A Paskevich, and G Ernst. *VerifyThis 2018: A Program Verification Competition*. Université Paris-Saclay Research Report, 2019
- 6 M Huisman, R Monahan, P Müller and E Poll. *VerifyThis 2016: A program verification competition* Enschede, the Netherlands: University of Twente, 2016
- 7 M Huisman, V Klebanov, R Monahan and M Tautschnig. *VerifyThis 2015* International Journal on Software Tools for Technology Transfer, Springer, 2017
- 8 M Huisman, R Monahan, P Müller, W Mostowski and M Ulbrich. *VerifyThis 2017: A Program Verification Competition* Karlsruhe Reports in Informatics, Karlsruhe Institute of Technology, 2017
- 9 M Huisman, V Klebanov and R Monahan. *VerifyThis 2012: A Program Verification Competition* International Journal on Software Tools for Technology Transfer, Springer, 2015

3.16 On Automatic Contract Synthesis

Philipp Rümmer (Uppsala University, SE)

License © Creative Commons BY 4.0 International license
© Philipp Rümmer

Joint work of Anoud Alshnakat, Dilian Gurov, Christian Lidström, Philipp Rümmer

Main reference Anoud Alshnakat, Dilian Gurov, Christian Lidström, Philipp Rümmer: “Constraint-Based Contract Inference for Deductive Verification”, pp. 149–176, Springer, 2020.

URL https://doi.org/10.1007/978-3-030-64354-6_6

This survey talk considers software contracts, consisting of pre-conditions and post-conditions, for imperative programs. Contracts of this kind enable modular verification of programs, and are therefore an important tool to scale up verification methods to large code bases. Formulating sufficient contracts is a time-consuming process, however, making the possibility to compute contracts automatically an attractive option. After studying the algebraic structure of software contracts, the talk surveys some of the contract inference methods that have been proposed in the literature. Considered contract inference methods derive contracts from programs implementations and program specifications, and can produce auxiliary annotations that complement manually written specifications.

The talk is partly based on recent joint work with Anoud Alshnakat, Jesper Amilon, Zafer Esen, Dilian Gurov, and Christian Lidström.

3.17 When You Hit the Frame

Thomas Santen (Formal Assurance – Aachen, DE)

License © Creative Commons BY 4.0 International license
© Thomas Santen

Frame conditions of the form $x' = x$ are essential for pre/post condition contracts to specify the parts of a state that remain unchanged under execution of an operation. Though seemingly trivial equations, frame conditions are a common source of error in specifications of industrial systems, in particular when they need to be propagated through several levels of abstraction. The talk discusses common yet expensive error patterns and an approach to debug erroneous frame specifications.

3.18 Correctness-by-Construction: The CorC Ecosystem

Ina Schaefer (KIT – Karlsruher Institut für Technologie, DE)

License © Creative Commons BY 4.0 International license
© Ina Schaefer

Joint work of Ina Schaefer, Tabea Bordis, Tobias Runge, Alexander Kittelmann

Main reference Tabea Bordis, Loek Cleophas, Alexander Kittelmann, Tobias Runge, Ina Schaefer, Bruce W. Watson: “Re-CorC-ing KeY: Correct-by-Construction Software Development Based on KeY”, in Proc. of the The Logic of Software. A Tasting Menu of Formal Methods – Essays Dedicated to Reiner Hähnle on the Occasion of His 60th Birthday, Lecture Notes in Computer Science, Vol. 13360, pp. 80–104, Springer, 2022.

URL https://doi.org/10.1007/978-3-031-08166-8_5

Correctness-by-Construction (CbC) is an incremental software development technique to create functionally correct programs guided by a specification. In contrast to post-hoc verification, where the specification and verification mainly take part after implementing a program, with CbC the specification is defined first, and then the program is successively created using a small set of refinement rules. This specification-first approach has the advantage that errors are likely to be detected earlier in the design process and can be tracked more easily. Even though the idea of CbC emerged many years ago, CbC is not widespread and is mostly used to create small algorithms. We believe in the idea of CbC and envision a scaled CbC approach that contributes to solving problems of modern software development. In this presentation, I will give an overview of our work on CbC in four different lines of research. For all of them, we provide tool support building the CorC ecosystem that even further enables CbC-based development for different fields of application and sizes of software systems.

3.19 VeyMont: Parallelising Verified Programs instead of Verifying Parallel Programs

Petra van den Bos (University of Twente – Enschede, NL)

License © Creative Commons BY 4.0 International license
© Petra van den Bos

Joint work of Petra van den Bos, Sung-Shik Jongmans

We present VeyMont: a deductive verification tool that aims to make reasoning about functional correctness and deadlock freedom of parallel programs (relatively complex) as easy as that of sequential programs (relatively simple). The novelty of VeyMont is that

it “inverts the workflow”: it supports a new method to parallelise verified programs, in contrast to existing methods to verify parallel programs. Inspired by methods for distributed systems, VeyMont targets coarse-grained parallelism among threads (i.e., whole-program parallelisation) instead of fine-grained parallelism among tasks (e.g., loop parallelisation).

4 Working groups

4.1 Two-state versus Trace-based Contracts: Report from Breakout Group 1

Dilian Gurov (KTH Royal Institute of Technology – Stockholm, SE), Bernhard Beckert (KIT – Karlsruher Institut für Technologie, DE), Alessandro Cimatti (Bruno Kessler Foundation – Trento, IT), Paula Herber (Universität Münster, DE), Srdan Krstic (ETH Zürich, CH), Martin Leucker (Universität Lübeck, DE), Christian Lidström (KTH Royal Institute of Technology – Stockholm, SE), Marco Scaletta (TU Darmstadt, DE)

License © Creative Commons BY 4.0 International license
 © Dilian Gurov, Bernhard Beckert, Alessandro Cimatti, Paula Herber, Srdan Krstic, Martin Leucker, Christian Lidström, and Marco Scaletta

Introduction

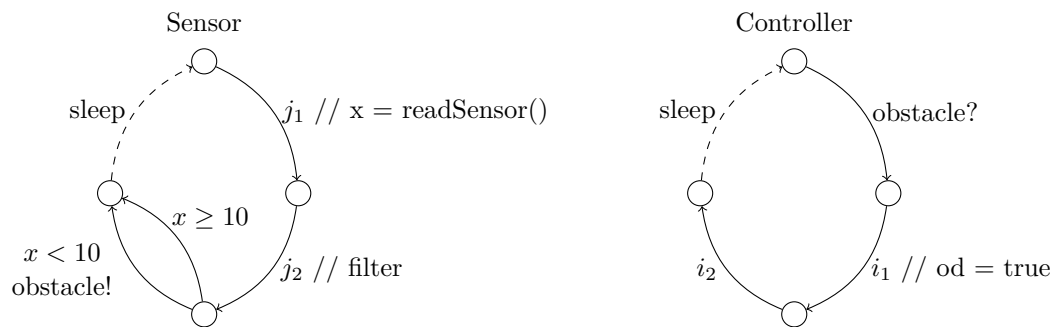
The original *design-by-contract* paradigm [6] has been proposed in the context of Hoare logic [4], which essentially relates initial with final states of a computation. While in Hoare logic, pre- and postconditions are usually used to put assertions on the initial and final state, Meyer used the broader concept of assumptions and guarantees. In this report, we refer to contracts that relate initial with final states as *two-state contracts*. In contrast, temporal properties are properties of sequences of states, called *traces*. Contracts about the temporal behaviour of components can therefore be termed *trace-based contracts*.

But when shall one use contracts of the first kind, and when of the second? Can they be combined? How can they be written, concretely, and with what formal semantics? How can they be formally verified, preferably automatically? These are some of the questions which Breakout Group 1 set out to explore.

Motivating Example

We discussed an example, where two interacting components, namely a *Sensor* and a *Controller* interact via an event, as shown with two simple finite state machines in Figure 1. In this example, a *Sensor* reads sensor values (action j_1) and performs some internal computations on them, e.g., filtering (j_2). Then, it sends a warning to the controller (*obstacle!* denotes sending event *obstacle*) if the read sensor value is below a given threshold ($x < 10$). Then, it goes to sleep for some time or until new sensor data is available. The controller runs concurrently with the sensor, and synchronizes on the event as a receiver (*obstacle?*). Whenever it receives an *obstacle* warning, it sets a local variable obstacle detected (*od*) to true, and performs some action i_2 , for example to avoid the obstacle.

Classical contract-based approaches encapsulate the local effects of a component with a contract. A challenge of dynamically interacting components is that we want to reason about global properties based on the local ones. For example, whenever an obstacle is



■ **Figure 1** Interacting Concurrent Components.

detected ($x < 10$), there actually is a reaction of the controller, which can be expressed with the CTL formula $AG(x < 10 \rightarrow AF od)$. This global property involves variables from both the controller and the sensor. To encapsulate the local effects of the processes, possible abstractions for this example might merge internal computations into atomic code blocks (e.g., $c_i = i_1, i_2$ and $c_j = j_1, j_2$) and describe the behavior of the processes with behavioral specifications, e.g., as regular expressions $(c_j \text{ obstacle!})^*$ or $(\text{obstacle? } c_i)^*$, as recently proposed in [3].

Such a behavioral specification enables us to reason about properties that result from the local effects of both processes in combination, e.g., that the actions in c_j are actually followed by actions in c_i , because the processes are synchronized via the *obstacle* event. However, whether the property actually holds depends on the concrete synchronization mechanism, and the analysis is further complicated if there runs a scheduler in the background, and we have asynchronous communication. Such behavioral specifications can be extended towards more complex properties; for example, a valve may open or close, take some time to open, might fail, might be scheduled and so forth, and intermediate steps might be relevant or not. We also discussed that it might be useful to have not just one contract, but a set of possible abstractions one can use for different purposes.

Two-state and Trace-based Contracts

To cope with complex, concurrent systems as shown above, a combination of two-state and trace-based contracts might be useful. We made the following observations and suggestions.

Contracts, Abstractly

At the most abstract level, a contract should specify the behaviour of a component along its *interface*. Conceptually, the *interface behaviour* of a component consists of the possible sequences of side-effects that invoking a component (e.g., procedure) gives rise to.

Two Aspects of Computation

There are conceptually two aspects of computation: *state transformation* and *interaction*. Two-state contracts are better suited, and have been designed, for the specification of the former, while trace-based contracts for the latter.

Specifying Contracts

Contracts are often stated as assumption-guarantee pairs. Assumptions can conceptually be of a temporal nature (such as, for example, in “a file should only be closed if it has been opened before that”), and could be specified in PLTL (past-time LTL), but could equivalently be expressed with a state formula, provided some “ghost” state is used to capture the “relevant part” of the history.

For specifying temporal contracts of procedural languages, logics of nested words may be useful (such as CaRet [1]), since they offer a construct that essentially amounts to the two-state contract corresponding to a procedure call.

Combining the Aspects

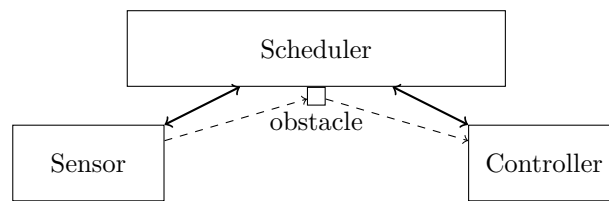
Complex systems typically exhibit both aspects of computation mentioned above. The aim of specification of such systems should be to abstract from the data transformation by *summarising* these with two-state contracts, and focus on the trace properties of the event sequences. We need methods and tools to help engineers to combine reasoning about data transformation with reasoning about event sequences (which, in separation, engineers know how to work with).

While trace-based properties are what we want to verify on a system level, two-state ones are rather on the local level and capture the concrete details of a component with finer granularity. Furthermore, the trace-based view should be used for interactions and interface specifications, and the two-state view for computations on data, i.e., state transformations. To connect these two views, contracts might be useful; for example, by means of two-state contracts one can *abstract* code blocks into TLA actions [5], and then apply model checking for the verification of temporal properties. This idea has recently been advocated in [2] by two of the present authors.

This idea could be embedded into a more general methodology for verification by building an abstract system view, as shown in Figure 2 for our motivating example. A preliminary approach to incorporate this idea into a verification process would be to perform three steps, considering the global and the local views separately:

1. Use interface specifications to get a trace-based system level view, while abstracting from the inner workings of each component; e.g., a behavioral specification of the overall system, as a composition of local views, could look for our example like this: $(c_j \text{ obs!})^* || (\text{obs? } c_i)^*$.
2. For each component: analyse locally, with a two-state view, under what assumptions a component adheres to its trace-based interface specification; e.g., as postcondition, our sensor has sent the obstacle warning under the condition $x < 10$ in the final state if, as precondition, it has initially been scheduled: $(\text{scheduled}, x < 10 \rightarrow \text{obstacle!})$.
3. Verify the global property on the behavioral system specification, while using assumptions and guarantees from the local two-state based views as abstractions of the internal behavior of components, e.g. check that $AG(x < 10 \rightarrow AF od)$ is satisfied.

While the first step should focus on processes and their interactions, the second step should mainly capture local effects as a two-state view. With that, a separation of concerns and modular verification might be possible even for complex systems.



■ **Figure 2** Abstract View with Contracts.

Conclusion

The discussion showed that contracts are used very differently in different communities. Depending on the interpretation of a contracts, and in particular, depending on whether a contract is interpreted as a two-state pair or a behavioral, trace-based specification, contracts play a very different role in program and system verification. Two-state contracts are very well-suited to capture local, transformational effects with no side-effects, while trace-based specifications are very well-suited to capture dynamic interactions and sequences of events.


We believe that *combining the two paradigms* has a high potential to modularize the verification effort in complex systems, and to increase the scalability and feasibility of formal methods. However, just throwing the two specification paradigms together might impede the feasibility of the underlying verification techniques, as the expressiveness of the contract language becomes too high. Instead, systematic abstractions and a separation of concerns are highly desirable.

References

- 1 Rajeev Alur, Kousha Etessami, and P. Madhusudan. A temporal logic of nested calls and returns. In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2004)*, volume 2988 of *Lecture Notes in Computer Science*, pages 467–481. Springer, 2004.
- 2 Jesper Amilon, Christian Lidström, and Dilian Gurov. Deductive verification based abstraction for software model checking. In *Leveraging Applications of Formal Methods, Verification and Validation (ISoLA 2022)*, volume 13701 of *Lecture Notes in Computer Science*, pages 7–28. Springer, 2022.
- 3 Gidon Ernst, Alexander Knapp, and Toby Murray. A hoare logic with regular behavioral specifications. In *Leveraging Applications of Formal Methods, Verification and Validation (ISoLA 2022)*, volume 13701 of *Lecture Notes in Computer Science*, pages 45–64. Springer, 2022.
- 4 C. A. R. Hoare. An axiomatic basis for computer programming. *Commun. ACM*, 12(10):576–580, 1969.
- 5 Leslie Lamport. The temporal logic of actions. *ACM Transactions on Programming Languages and Systems*, 16(3):872–923, 1994.
- 6 Bertrand Meyer. Applying “Design by contract”. *IEEE Computer*, 25(10):40–51, 1992.

4.2 Abstraction in Contracts: Report from Breakout Group 2

Reiner Hähnle (TU Darmstadt, DE), Asmae Heydari Tabar (TU Darmstadt, DE), Eduard Kamburjan (University of Oslo, NO), Nikolai Kosmatov (Thales Research & Technology – Palaiseau, FR), Rosemary Monahan (National University of Ireland – Maynooth, IE), Thomas Santen (Formal Assurance – Aachen, DE), Ina Schaefer (KIT – Karlsruher Institut für Technologie, DE), Julien Signoles (CEA LIST – Gif-sur-Yvette, FR), Alexander J. Summers (University of British Columbia – Vancouver, CA)

License  Creative Commons BY 4.0 International license
 © Reiner Hähnle, Asmae Heydari Tabar, Eduard Kamburjan, Nikolai Kosmatov, Rosemary Monahan, Thomas Santen, Ina Schaefer, Julien Signoles, and Alexander J. Summers

Motivation

During the plenary discussions in the seminar it was universally agreed that the ability to specify data structures at a suitably abstract, implementation-independent level is crucial to keep specification and verification effort of complex programs manageable. At the same time, the abstraction mechanisms offered by contemporary contract languages are insufficient to achieve this goal.

In addition to feasibility of the specification and verification effort, the capability to abstract away from implementation details in specifications has further important advantages: (i) Better communication with stakeholders at their level of abstraction, (ii) encouragement to write abstract specifications first, (iii) independence from implementation language (multi-language support), (iv) independence from a specific implementation.

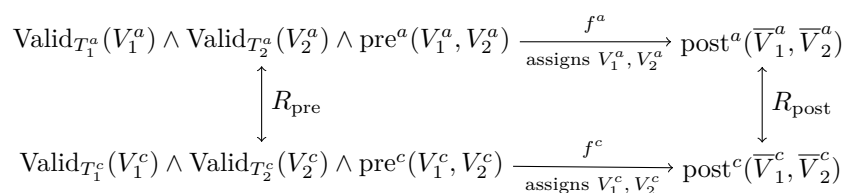
In consequence, the breakout group was tasked to explore what kind of abstraction mechanisms might be required and suitable.

Problem Description

Consider a piece of client code that intends to use the implementation of a given API. For example, the client needs to manage a *set of objects* of a certain type. Most languages offer an API for managing collections of objects, including sets. For efficiency reasons, sets are implemented with mutable data structures, such as arrays or linked lists. But this makes it problematic to specify the intended behavior at the *abstract* level, i.e. in terms of *sets*, because separation properties are specified at the implementation level and also differ among different implementations. In addition, one must make sure that the usage of the API by the client does not break memory separation.

In consequence, we need a discipline and mechanisms that encapsulate separation properties and that ensure their preservation at an abstract level. Current mechanisms as realized in languages such as Dafny or JML are based on *represents* clauses that define how abstract specification elements can be implemented in terms of concrete ones. While it is certainly essential to provide such representations, this is not sufficient to achieve abstraction in the sense outlined above for a number of reasons:

- The relation between abstract elements and the read/write footprint of concrete ones is not explicit, which is necessary to compute and verify assumptions.
- Represents clauses that relate to the same abstract datatype are not grouped together and named.
- There is no control over the visibility of the implementing elements.
- There are no specification constructs that permit to express under which conditions a given property of an abstract specification is preserved under a given operation.



■ **Figure 3** Diagram for verification of concrete implementation with the help of abstract types.

Sketch of a Proposal

In consequence of the above considerations, we propose the concept of an *abstract type*.¹ This is not a type of the underlying implementation language and outside its type hierarchy. Rather, we are thinking of a trait-like structure that might look as follows:

```

abstract type  $T^a \langle P \rangle$  {
  // 1. representation of type instance, invariants
  // 2. pure operations with representation
  // 3. impure operations with representation
  // 4. declare abstract specification building blocks:
  //     opaque, but extensible
}

```

This defines an abstract type T^a parameterized with some object type P (which we omit in the following for brevity). It provides

1. the representation of a T^a object in terms of concrete *implementing types* T_1^c, \dots, T_n^c , together with invariants that define what a *valid* instance is, for example, the property that a set is idempotent. The predicate $\text{Valid}_{T^a}(V^a)$ expresses that V^a is a valid value of T^a .
2. a set of pure operations that do not change the state and which can be used in assertions.
3. a set of impure operations.
4. a subset of pure and impure operations to be used as building blocks in abstract specifications. The remaining operations and the definitions are opaque.

Let us call the relation between abstract and concrete types as sketched above a *coupling*. We will also need to define *abstract properties* that relate one or more abstract object:

```

abstract property  $\text{prop}(T_1^a, \dots, T_n^a)$  {
  // representation in terms of concrete properties;
  // invariants
}

```

Abstract properties are domain-specific and should be defined in connection with a given abstract type. Examples of properties are: separation, reachability, order, containment, etc.

We illustrate how this setup can be used to simplify verification of concrete types, i.e. the *implementation* of an abstract type. Consider the bottom row in Fig. 3, representing the following contract of a concrete implementation of function f^c :

¹ Their might be a better terminology, this is just a working proposal.

```

 $f^c$  {
  requires   Valid $_{T_1^c}(V_1^c) \wedge$  Valid $_{T_2^c}(V_2^c) \wedge$  pre $^c(V_1^c, V_2^c)$ ;
  ensures    post $^c(\overline{V}_1^c, \overline{V}_2^c)$ ;
  assignable { $V_1^c, V_2^c$ };
}

```

Assume we want to prove this contract without referring to the concrete implementation as much as possible. We can assume that the corresponding *abstract* contract for f^a (analogous to f^c) has been shown already. Moreover, we can assume the coupling relation

$$R_{\text{pre}} \equiv \bigwedge_{i=1,2} (\text{Valid}_{T_i^a}(V_i^a) \wedge \text{Valid}_{T_i^c}(V_i^c) \wedge [\text{coupling of } V_i^a, V_i^c \text{ via } T_i^a])$$

to be given. The corresponding relation R_{post} can be computed using the implementation of f^c .

Now we establish the preservation property for pre and post:

$$R_{\text{pre}} \longrightarrow (\text{pre}^a(V_1^a, V_2^a) \longleftrightarrow \text{pre}^c(V_1^c, V_2^c)) \quad (1)$$

$$R_{\text{post}} \longrightarrow (\text{post}^a(\overline{V}_1^a, \overline{V}_2^a) \longleftrightarrow \text{post}^c(\overline{V}_1^c, \overline{V}_2^c)) \quad (2)$$

If the properties are chosen well and with the help of some composition theorems, this needs to be shown only once.

This should be sufficient to establish the concrete contract of f^c , i.e. the following should be provable as a meta theorem:

$$R_{\text{pre}} \wedge R_{\text{post}} \wedge \text{pre}^c(V_1^c, V_2^c) \longrightarrow \text{post}^c(\overline{V}_1^c, \overline{V}_2^c) .$$

Conclusion and Open Questions

Obviously, this is just a sketch, not a formal proof. To go further, we need to instantiate the schema for specific frameworks and look at a concrete example.² Notably, it must be clarified in which way our proposal generalizes model fields and methods. The meta theorem(s) must be formally proven, the assumptions clarified, etc.

Besides this, there are several open questions:

- What are minimal assumptions to make this work? Can the equivalence in eqs. (1), (2) be relaxed?
- What are the links to the B-method, as well to abstract interpretation?
- We assumed that the footprints of the V_i^c are disjoint – can it be generalized?
- To be practicable, the framework sketched here should be supported by specification idioms and patterns.

² During the breakout sessions, Alexander Summers sketched how a similar setup might look in the VIPER framework.

4.3 Specification Engineering: Report from Breakout Group 3

Wolfgang Ahrendt (Chalmers University of Technology – Göteborg, SE), David Cok (Safer Software Consulting – Rochester, US), Gidon Ernst (LMU München, DE), Sophie Lathouwers (University of Twente – Enschede, NL), Giles Reger (University of Manchester, GB), Kristin Yvonne Rozier (Iowa State University – Ames, US), Philipp Rümmer (Uppsala University, SE)

License © Creative Commons BY 4.0 International license
© Wolfgang Ahrendt, David Cok, Gidon Ernst, Sophie Lathouwers, Giles Reger, Kristin Yvonne Rozier, and Philipp Rümmer

We document the outcome of the break-out sessions on Specification Engineering, held at the Dagstuhl Seminar on Principles of Contract Languages, November 2022.

Research Question

We identify two main activities in engineering the (formal) specification of a system component or unit: a) the coming about of some first version of a specification, and b) developing existing specifications further, by maintenance, correction, refinement, extension, and alike. Activity a) refers to the initial formalization of the component’s requirements, which could be based on natural language descriptions, on the specifiers own understanding of the requirements, or other sources. While this is a very relevant problem, it has not been the focus of our discussions, and is not what this summary contributes to. Instead, we focus entirely on b), i.e., the activity on assessing the quality of a given status of the specification, and developing it further accordingly.

Specifications ought to represent certain desired properties of the system, typically as constraints, postulates, and models of the behavior. Those should be *more obviously correct than the code itself*, and should *sufficiently describe the guarantees that other components rely on*.

Large parts of the realm of formal methods is devoted to providing formal evidence that a system or component actually implements its specification, through mechanized verification. As of yet, much less focus is given to providing evidence on the correspondence between intended requirements and the formal specification, even if there is high demand on addressing this issue. Accordingly, we are interested in *the correspondence between the mental model inside the head of developers and the model that is actually formalized as by the specification*. Moreover, we are interested in *the correspondence between the what properties other components require, and what the specification of this component actually guarantees*.

While there are some established techniques, e.g. from the area requirements validation such as illustrating the meaning of temporal logic formulas, we seek to integrate these into a general framework that supports a systematic tool-driven process to ensure the aforementioned correspondences. Therefore, our research goal is

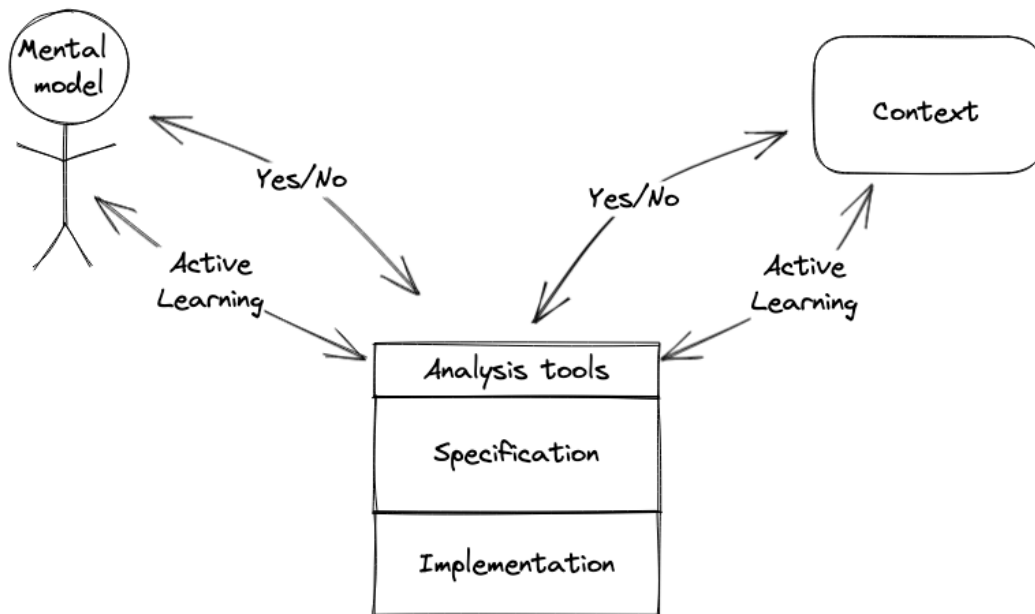
- to bridge between the mental model of the system’s *intended* behavior and the *formalized* specification, and
- to bridge between requirements *assumed* by other components, and the *guarantees* provided by this specification.

We have identified the following main questions.

- How is the process driven? E.g., which (kinds of) questions should the developer be able to pose to the system’s specification to get a better understanding, and similarly, which questions should be derived from the specification to be posed to the developer?
- What are the different dimensions and artifacts over which this dialogue is formulated and how can we ensure that these have a formally precise meaning while at the same time being easy to understand?
- What tools exist today that can be leveraged at which stages of the process?
- What tools (or tool functionalities) are yet to be provided by the community in the course of realizing such a process?

A method for specifications evolution

To address the bridge between the mental model of the system’s *intended* behavior and the *formalized* specification, we propose to use an iterative approach similar to active learning, see Figure 4.



■ **Figure 4** Active learning approach to refine specifications where different tools and techniques are used to iteratively improve the existing specification.

The main idea behind this approach is that the user can iteratively improve upon the existing specification by using different tools and techniques to gain a better understanding of the current specification, and develop it further accordingly. Similarly, the context in which the system is used can use a similar approach to enforce limitations of the system’s context. Focusing on the relation of the user’s mental model and the specification for now, the approach caters for tool supported querying in two directions. In the one direction, the user can post queries, to be answered by tools analyzing the specification. These answers shall help the user to evaluate how well the specification matches her own mental model of the component. In the other direction, tools can generate queries to the user, for instance by deducing consequences of the specification. The user can “answer” the query by indicating

whether the that consequence matches the intention. Also, tools can question the use of a common logical fallacies, anti-patterns, inconsistencies or vacuously true parts of the specification.

When asking questions to the user in this setting, it is important to take a few things into account. Firstly, one should ask the most important questions first. To achieve this one could use dynamic question prioritization. Secondly, one should be efficient with the questions that are asked to avoid overburdening the user. Finally, one should ask both positive and negative questions.

Techniques and tools for specification analysis

Some examples of techniques that can be used in this active learning setting to gain a better understanding of the specification are exemplified below. These techniques are well-known and their integration into our method will benefit from advances in the respective research areas.

Generating models of specifications helps to make concrete what specifications mean. The existence of models in the first place rules out vacuous specifications, i.e., formalizations that admit no behavior at all (e.g. an unsatisfiable precondition renders a method contract vacuously true) or that trivially allow all behaviors. Such models have a wide variety of uses cases, cf. below, and can be generated with the help of modern SAT/SMT solvers.

Visualization is a powerful tool that helps to illustrate the meaning of specifications. This has notably been explored for temporal logic formulas, where e.g. input signals for cyber-physical systems can be generated automatically and presented side-by-side with the corresponding outputs. Such a visual presentation has the advantage that it is very easy for a human to assess whether it matches their mental model.

The original formulation of the specification as provided by the engineer can be transformed symbolically and *re-phrased* by automated tools. Examples are the simplification and normalization of formulas and the computation of logical consequences and lemmas via theory exploration. This help to clarify what the specification means in the general sense, in contrast to examples which can only cover concrete cases.

Linter-like checks can be used to discover typical formalization mistakes. For instance, a universal quantifier should always be paired with an implication and an existential quantifier should be paired with a conjunction, as an example, the formula $\exists x. x \in P \implies Q(x)$ almost certainly does not express the engineer’s intention. Such checks can be integrated with a database of typical specification patterns and anti-patterns. Another example are typical liveness and safety properties in temporal logic.

Mutation testing is technique that attempts to discover the effect of changing the specification in certain “typical” ways (e.g. to replace an occurrence of $<$ by \leq). In software testing this is applied to assess the robustness of test-suites, here it can similarly be used to discover whether such mutations are in conflict with some insights learned already as evidence that the correspondence between the mental model and the specification has been exercised in sufficient detail to notice the mutation at all.

This approach can be used for many different types of specifications, e.g. pre-/postcondition style contracts, temporal logic formulas, or test cases. Depending on the type of specification that you are interested in, different techniques will be available in this iterative approach.

The Way Forward

In order to move forward in the direction of realizing the method sketched above, one should survey existing tools for analyzing specifications and answering queries about them. Equally important would be an investigation on what kind of queries (and their according answers) would help users to judge the extent to which a current specification matches the mental model.

A proof of concept for the proposed method could be to instantiate the approach with at least two concrete ways, e.g., with two tools supporting to query pre/post-specifications or trace-based specifications, respectively. Those shall then be applied to a few small case studies to begin with. This will help to demonstrate how current methods and tools can already support such a process and where gaps and weaknesses are.

4.4 Interoperable Contracts: Report from Breakout Group 4

Petra van den Bos (University of Twente – Enschede, NL), Claire Dross (AdaCore – Paris, FR), Sylvain Hallé (University of Quebec at Chicoutimi, CA), Marieke Huisman (University of Twente – Enschede, NL), Marie-Christine Jakobs (TU Darmstadt, DE), Mattias Ulbrich (KIT – Karlsruher Institut für Technologie, DE)

License © Creative Commons BY 4.0 International license
© Petra van den Bos, Claire Dross, Sylvain Hallé, Marieke Huisman, Marie-Christine Jakobs, and Mattias Ulbrich

This document summarises the discussions of the break out session on Interoperable Contracts held at the Dagstuhl Seminar on Principles of Contract Languages.

Research Question

The current status in research on contracts is that there is a versatile set of tools for verifying software, but these tools cannot be used interchangeably. We explore what would be needed to achieve interoperability. Specifically, our research question is:

What contract infrastructure is needed to allow tools to interoperate w.r.t. formal contracts?

We consider this question for a wide range of tools. Besides tools using pre- and postcondition contracts, we also include tools with other notions of contracts for software. A motivation for including these tools is that they also target verification of software. In Table 1 we list the research areas of the type of tools we consider. For each area, we state the most important ‘types’ of contract, i.e. what needs to be specified to enable the software verification. Additionally, the table lists the mode of operation: is software analysed statically, or at run-time, so dynamically.

From the overview of Table 1 we selected two specific interoperability cases:

Pre-postcondition contracts only Interoperability for two deductive verification tools, or a deductive verification and a run-time assertion checking tool, using pre- and postcondition contracts as specification language.

Different contract types + static-dynamic Interoperability between a deductive verification tool and a dynamic verification tool using automata and/or temporal logic specifications.

■ **Table 1** Research areas of software verification tools.

Research area	Included contract types	Mode
Deductive verification	Pre- and postconditions, invariants	Static
Software model checking	Reachable code points, pre- and postconditions, assertions	Static
Classical model checking	Automata, temporal logic	Static
Run-time assertion checking	Pre- and postconditions, assertions	Dynamic
Run-time verification	Automata, temporal logic, regexes	Dynamic
Automated testing	Automata	Dynamic

Interoperability between Tools using Pre-Postcondition Contracts

Next, we enumerate the main challenges that we identified for interoperability between deductive verification tools and run-time assertion checkers, using pre- and postcondition contracts as specification language.

Contract language and translation. Tools use languages with different syntax and semantics.

Translation from one tool language to the other is needed to make tools interoperable. A promising direction is to design a common or unified language for contract exchange, though this could be “too late now” with the current diverse landscape.

Executable static languages. There are tools with languages that can be used for both static and dynamic verification of pre-postcondition contracts, e.g. the E-ACSL language component of ACSL in tool Frama-C, and the language of the tool Spark have been designed such that all specifications can be used for static verification, but are also executable for run-time verification. However, this requires that the static and dynamic semantics of the assertions coincide, which makes it challenging.

Implicit assumptions. When exchanging verification tasks, implicit assumptions need to be taken into account, as an assumption can change the semantics of a contract. We distinguish three types of differences in assumptions:

- Differences in semantics that do not cause any issue for the combination. For example, one tool does not verify exceptional behaviour like run-time exceptions, so it must be ensured that this part is verified by another tool that is able to do this.
- Differences that need to be made explicit in the pre/postcondition, but still allow the interaction to take place. For example, if one tool assumes that objects are non-null, then an explicit assertion can be used in another tool without this assumption.
- Different constructs that must be disallowed because they are handled in a different way by the two tools. For example, different semantics for floating point numbers are used in two different tools, so the verification result is different for the two tools.

Division of the verification task. To be able to use the interoperability of tools, we need a method for dividing the verification task. Here, the contract, the code, or both could be divided in pieces for distribution to different tools.

Coordinating tool interoperability. It needs to be decided, either manually or automatically, which tool verifies what. Note here that to be able to use different tools effectively, an explicit overview or mapping of tool capabilities is needed, to know which parts or properties need to be verified by which tool for obtaining the most optimal (e.g. most conclusive or most time-efficient) verification result.

Combining verification results. The verification results of the used tools need to be combined to one global verification result saying what was (not) proven. Used assumptions need to be included in this result. Also, in case of verification failures, it must be possible to trace back the failures for analysis and modification purposes.

Application to multi-language software. A software system may consist of multiple languages. Verifying this with a single verification tool is already challenging.

Static-Dynamic Interoperability for Different Contract Types

Below we list approaches for interoperability between a tool using contracts, and a tool using automata and/or temporal logic. We note that most approaches can be found in the existing literature and that static-dynamic interoperability may be coordinated in both directions.

Outsourcing. A deductive verifier could outsource difficult-to-prove conditions to a (runtime) monitor or assertion checker, or a runtime monitor could outsource difficult-to-monitor conditions to a deductive verifier.

Monitor reduction. Deductive verification or model checking can be used to check properties such that monitoring assertions can be removed. For example, one might omit monitoring a part of a temporal logic formula because it has been proven via static verification.

Contracts for abstraction hierarchy. One may replace a subcomponent of an automaton by a pre- post condition contract or a temporal logic formula and use a model checker to verify this abstraction step.

Contracts for automaton actions. One may use an automaton to model the sequences of method calls, where a method corresponds to a transition, specified by a contract: the precondition is the guard of the transition, and the postcondition the action or resulting state change.

Correspondence checks between automaton and code via annotations. One may translate an automaton to annotations in the program to prove correspondence between the automaton and the code.

Verification of unbounded data in testing. One may use deductive verification to abstract from unbounded data parameters in an automaton used for test generation.

Follow-Up Plans

Several of the participants in the working group would like to explore the interoperability of tools that use pre- and postconditions as contracts more. They plan to discuss more about these different categories of implicit assumptions, and whether these could be identified in a systematic manner.

For the combination of static and dynamic verification techniques with different contract types, the results were more inconclusive. It is an interesting area, with a high potential, but we need further discussions to come up with concrete new ideas on how to exploit this combination.

Participants

- Wolfgang Ahrendt
Chalmers University of
Technology – Göteborg, SE
- Bernhard Beckert
KIT – Karlsruher Institut für
Technologie, DE
- Alessandro Cimatti
Bruno Kessler Foundation –
Trento, IT
- David Cok
Safer Software Consulting –
Rochester, US
- Claire Dross
AdaCore – Paris, FR
- Gidon Ernst
LMU München, DE
- Dilian Gurov
KTH Royal Institute of
Technology – Stockholm, SE
- Reiner Hähnle
TU Darmstadt, DE
- Sylvain Hallé
University of Quebec at
Chicoutimi, CA
- Paula Herber
Universität Münster, DE
- Asmae Heydari Tabar
TU Darmstadt, DE
- Marieke Huisman
University of Twente –
Enschede, NL
- Marie-Christine Jakobs
TU Darmstadt, DE
- Eduard Kamburjan
University of Oslo, NO
- Nikolai Kosmatov
Thales Research & Technology –
Palaiseau, FR
- Srdan Krstic
ETH Zürich, CH
- Sophie Lathouwers
University of Twente –
Enschede, NL
- Martin Leucker
Universität Lübeck, DE
- Christian Lidström
KTH Royal Institute of
Technology – Stockholm, SE
- Rosemary Monahan
National University of Ireland –
Maynooth, IE
- Doron A. Peled
Bar-Ilan University –
Ramat Gan, IL
- Giles Reger
University of Manchester, GB
- Kristin Yvonne Rozier
Iowa State University –
Ames, US
- Philipp Rümmer
Uppsala University, SE
- Thomas Santen
Formal Assurance – Aachen, DE
- Marco Scaletta
TU Darmstadt, DE
- Ina Schaefer
KIT – Karlsruher Institut für
Technologie, DE
- Julien Signoles
CEA LIST – Gif-sur-Yvette, FR
- Alexander J. Summers
University of British Columbia –
Vancouver, CA
- Mattias Ulbrich
KIT – Karlsruher Institut für
Technologie, DE
- Petra van den Bos
University of Twente –
Enschede, NL



Computational Social Dynamics

Martin Hoefer^{*1}, Sigal Oren^{*2}, Roger Wattenhofer^{*3}, and
Giovanna Varricchio^{†4}

- 1 Goethe-Universität – Frankfurt am Main, DE. mhoefer@em.uni-frankfurt.de
- 2 Ben Gurion University – Beer Sheva, IL. sigal3@gmail.com
- 3 ETH Zürich, CH. wattenhofer@ethz.ch
- 4 Goethe-Universität – Frankfurt am Main, DE. Varricchio@em.uni-frankfurt.de

Abstract

This report documents the program and outcomes of Dagstuhl Seminar 22452 “Computational Social Dynamics”. The seminar addressed social and dynamic problems in the field of algorithmic game theory, and their implications in numerous applications, such as fair division, financial networks, or behavioral game theory. We summarize organizational aspects of the seminar, the talk abstracts, and the problems that were discussed in the open problem sessions.

Seminar November 6–11, 2022 – <http://www.dagstuhl.de/22452>

2012 ACM Subject Classification Theory of computation → Algorithmic game theory and mechanism design; Information systems → Social networks; Theory of computation → Theory and algorithms for application domains; Theory of computation

Keywords and phrases algorithmic game theory, behavioral economics, fair division, financial networks, social networks

Digital Object Identifier 10.4230/DagRep.12.11.28

1 Executive Summary

Martin Hoefer (Goethe-Universität – Frankfurt am Main, DE)

Sigal Oren (Ben Gurion University – Beer Sheva, IL)

Roger Wattenhofer (ETH Zürich, CH)

License  Creative Commons BY 4.0 International license
© Martin Hoefer, Sigal Oren, and Roger Wattenhofer

Algorithmic techniques provide a powerful toolbox for understanding many phenomena arising in modern society. Often, these phenomena are related to dynamics (e.g., dynamic information spreading, or dynamics in social networks that result from agent interaction). A large part of the present social interaction on networks can be expressed using game-theoretic or microeconomic concepts, e.g., the dynamics of opinions in networks, pricing problems and viral marketing, network-based effects of opinions, group formation, cognitive bias, etc. These problems are rigorously analyzed in the area of algorithmic game theory, where researchers apply the algorithmic toolbox to analyze various social systems.

In this field, a number of applications have not received sufficient attention, which are recently becoming increasingly prominent. For example,

- issues of fairness and bias are central challenges in modern societies.
- behavioral economists are challenging standard assumptions that humans are maximizing expected utility. This change in perspective also poses new challenges for suitable models and algorithm design.

* Editor / Organizer

† Editorial Assistant / Collector



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

Computational Social Dynamics, *Dagstuhl Reports*, Vol. 12, Issue 11, pp. 28–44

Editors: Martin Hoefer, Sigal Oren, Roger Wattenhofer, and Giovanna Varricchio



Dagstuhl Reports

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

- an important trend in the modern economy are informational challenges, e.g., in problems involving recommendation, persuasion, delegation, or (smart) contract design.

The main aim of this seminar was to bring together a leading set of researchers to discuss these and other challenges, and to advance the state of the art in several new directions. The majority of participants were academics from computer science departments; some were from other disciplines such as economics, mathematics, or electrical engineering. All participants had strong interdisciplinary interests that typically span economics, game theory, and theoretical computer science.

The seminar started on Monday with an introductory session, in which participants introduced their name, affiliation, main research interest as well as a “crazy idea” or “provoking thought”. This session was very well-received by the participants, and it initiated discussions directly from the start. The subsequent program included four invited talks/tutorials of roughly one hour each, on several issues chosen by the organizers.

Monday Caragiannis talked about notions of fairness for resource allocation problems.

Tuesday Feldman presented recent innovations in algorithms for contract design.

Wednesday Plonsky discussed behavioral experiments and ideas to predict human behavior.

Thursday Wattenhofer surveyed issues in e-government, blockchains, and finance.

The contributed talks were solicited from the participants and lasted around 20-25 minutes each. In many of these talks there were lively discussions. They continued in the times from lunch to afternoon coffee, which were free for research and individual collaborative meetings.

There was a substantial set of presenters who focused on aspects of fair division. Rathi discussed epistemic notions of envy-freeness and their efficient computation. Bernadé presented results on online algorithms for fair division and guaranteeing notions of envy-freeness and Pareto-efficiency. Zick showed existence and efficient computation for several fairness criteria for matroid-rank valuations. Varricchio discussed algorithmic aspects of randomization and entitlements in fairness notions. Reiffenhäuser presented results on the quality of equilibria in fair division when strategic misreporting of preferences is allowed.

Another focus was the analysis of dynamics and stability in networks. Ventre discussed the clearing problem in financial networks with credit default swaps and showed FIXP-completeness results. Wilhelmi presented a game-theoretic model for clearing and results on equilibrium computation. Schmand considered a game-theoretic model for opinion formation in networks and convergence of natural best-response dynamics. Lenzner surveyed recent work on network creation games as well as network games that model segregation aspects.

Further talks addressed a number of different areas. Witkowski studied incentive-compatible forecasting competitions. Lavi considered incentives when several of these contests are available to participants. Klimm surveyed work on impartial selection, e.g., when agents have to decide on a representative member among themselves. Leyton-Brown posed new directions and open problems in the understanding of behavior in mobile gaming. Babichenko discussed aggregation mechanisms for anonymous information. Markakis presented a novel algorithm to compute an equilibrium in 2-player zero-sum games. Melnyk explained issues arising from Byzantine behavior in social choice. Last, but not least, Christodoulou gave a proof of the 23-year old Nisan-Ronen conjecture that every truthful mechanism for unrelated machine scheduling can only guarantee a linear approximation ratio.

The seminar was a big success. We believe it will stimulate new and very fruitful collaborations. We got laudatory feedback from many participants which is also reflected in the survey conducted by Dagstuhl.

We thank Giovanna Varricchio for serving as collector of abstracts and open problems.

2 Table of Contents

Executive Summary

<i>Martin Hoefer, Sigal Oren, and Roger Wattenhofer</i>	28
---	----

Overview of Talks


Regret-minimizing aggregation of anonymous information <i>Yakov Babichenko</i>	32
Dynamic fair division with partial information <i>Gerdus Benade</i>	32
Fairness in allocation problems (Tutorial) <i>Ioannis Caragiannis</i>	33
A proof of the Nisan-Ronen conjecture <i>Giorgos Christodoulou</i>	33
Algorithmic Contract Design (Tutorial) <i>Michal Feldman</i>	33
Impartial selection problems <i>Max Klimm</i>	33
From Monopoly to Competition: When Do Optimal Contests Prevail? <i>Ron Lavi</i>	34
Dynamics in Network Creation and Residential Segregation <i>Pascal Lenzner</i>	34
Mobile Games: An Exciting New Domain for Behavioral Game Theory <i>Kevin Leyton-Brown</i>	35
On new variants of multiplicative weights update and mirror prox methods for 0-sum games <i>Vangelis Markakis</i>	35
Byzantine Adversaries in Social Choice <i>Darya Melnyk</i>	36
Predictably Irrational? Can behavioral decision making research help predict human choice? (Tutorial) <i>Ori Plonsky</i>	36
New Fairness notions for Resource Allocation <i>Nidhi Rathi</i>	36
Allocating Indivisible Goods to Strategic Agents: Pure Nash Equilibria and Fairness <i>Rebecca Reiffenhäuser</i>	37
Asynchronous Opinion Dynamics in Social Networks <i>Daniel Schmand</i>	37
Best of Both Worlds: Agents with Entitlements <i>Giovanna Varricchio</i>	38
Systemic risk in financial networks – a computational perspective <i>Carmine Ventre</i>	39

Does Computation Change Society? (Tutorial)	
<i>Roger Wattenhofer</i>	39
Seniorities and Minimal Clearing in Financial Network Games	
<i>Lisa Wilhelm</i>	40
Incentive-Compatible Forecasting Competitions	
<i>Jens Witkowski</i>	40
A Simple Vision for Fair Division	
<i>Yair Zick</i>	41
Open problems	
Condorcet outcome for m independent binary issues	
<i>Andrei Constantinescu</i>	41
Interval cover for spatial crowdsourcing – Improving approximation	
<i>Vangelis Markakis</i>	41
Complexity of Public Goods Games on Graph – Nash equilibrium existence (Solved)	
<i>Noam Nisan</i>	42
Optimizing over Serial Dictatorships – Lower bound for optimal matching	
<i>Nidhi Rathi</i>	42
Altruistic Hedonic Games (Short talk)	
<i>Jörg Rothe</i>	43
Tractability frontier of justice criteria under submodular valuations	
<i>Yair Zick</i>	43
Participants	44

3 Overview of Talks

3.1 Regret-minimizing aggregation of anonymous information

Yakov Babichenko (Technion – Haifa, IL)

License  Creative Commons BY 4.0 International license
© Yakov Babichenko

Joint work of Itai Arieli, Yakov Babichenko, Inbal Talgam-Cohen, Konstantin Zabarniy

We study a model in which a decision maker aggregates information obtained from several symmetric agents. Each agent provides the decision maker with a recommendation about a binary state of nature, where the state is drawn from a known prior distribution. While the decision maker knows the marginal distribution of each agent’s recommendation, the correlation between the recommendations is chosen adversarially. The decision maker’s goal is to choose an information aggregation function minimizing the regret – the difference between her own mistake probability when guessing the state of nature, and the mistake probability of a Bayesian decision maker knowing the correlation between the recommendations.

We provide a characterization of the minimal regret for any number of agents as the maximal Jensen gap of a convex function that captures the probability of a correct guess by a hypothetical Bayesian decision maker. For a large number of agents, we deduce that apart from some borderline cases, the unique optimal aggregation function is the random dictator rule that chooses an agent uniformly at random and adopts her recommendation.

3.2 Dynamic fair division with partial information

Gerdus Benade (Boston University, US)


License  Creative Commons BY 4.0 International license
© Gerdus Benade

We consider the problem of fairly and efficiently allocating T indivisible items among n agents with additive preferences. Items arrive over a sequence of rounds, and must be allocated immediately and irrevocably before the next one arrives. When the agents’ valuations for the items are drawn from known distributions, it is possible to find allocations that are envy-free with high probability and Pareto efficient ex-post.

We study a partial-information setting, where values are drawn from unknown distributions and it is possible to elicit ordinal but not cardinal information. When a new item arrives, the algorithm can query each agent for the relative rank of this item with respect to a subset of the past items. We provide algorithms with strong simultaneous fairness and efficiency guarantees even with minimally expressive queries that ask for a comparison to a single previous item and show they are asymptotically optimal.

3.3 Fairness in allocation problems (Tutorial)


Ioannis Caragiannis (Aarhus University, DK)

License  Creative Commons BY 4.0 International license
© Ioannis Caragiannis

We will present variations of the problem of allocating indivisible items to agents with (mainly additive) valuations for the items. We will define basic fairness concepts such as proportionality and envy-freeness and discuss their basic properties. Next, we will introduce approximate versions of these concepts, such as envy-freeness up to some/any item (EF1/EFX) and maximin share fairness (MMS). We will present examples and many open problems.

3.4 A proof of the Nisan-Ronen conjecture


Giorgos Christodoulou (Aristotle University of Thessaloniki, GR)

License  Creative Commons BY 4.0 International license
© Giorgos Christodoulou

Noam Nisan and Amir Ronen conjectured that the best approximation ratio of deterministic truthful mechanisms for makespan-minimization for n unrelated machines is n . This work validates the conjecture.

3.5 Algorithmic Contract Design (Tutorial)

Michal Feldman (Tel Aviv University, IL)

License  Creative Commons BY 4.0 International license
© Michal Feldman
Joint work of Tomer Ezra, Paul Duetting, Michal Feldman, Thomas Kesselheim

Up until recently, Algorithmic Game Theory has mainly focused on the design of mechanisms that incentivize agents to truthfully report their private preferences. However, algorithms and incentives interact in many additional ways; the design of contracts being a prime example. While mechanism design deals with hidden preferences, contract design deals with hidden actions, and studies how best to incentivize agents to take costly actions, when their actions are hidden from the principal. With the transition of classic applications of contracts into computational platforms, algorithm design for such applications becomes timely and relevant.

In this talk, I will survey two papers on combinatorial contracts, which highlight different sources of complexity that arise in contract design.

3.6 Impartial selection problems

Max Klimm (TU Berlin, DE)

License  Creative Commons BY 4.0 International license
© Max Klimm

Impartial selection problems are concerned with situations where a group of agents selects a subset of the agents based on nominations from within the set. The fact that the agents act both as voters and as nominees may give rise to incentive issues when some agents may not be

willing to communicate their true opinion about who should be selected in order to influence their own chances of being selected. These issues may arise in a number of applications such as voting in committees, peer review in conferences where committee members also submit papers, and peer grading. One way to circumvent these incentive issues is to use impartial mechanisms that have the property that the selection probability of each agent is independent of its nominations. In this talk, I will give an overview on impartial selection mechanisms and point to some open problems in this area.

3.7 From Monopoly to Competition: When Do Optimal Contests Prevail?

Ron Lavi (University of Bath, GB)

License © Creative Commons BY 4.0 International license
© Ron Lavi

Joint work of Xiaotie Deng, Yotam Gafni, Ron Lavi, Tao Lin, Hongyi Ling

Main reference Xiaotie Deng, Yotam Gafni, Ron Lavi, Tao Lin, Hongyi Ling: “From Monopoly to Competition: Optimal Contests Prevail”, CoRR, Vol. abs/2107.13363, 2021.

URL <https://arxiv.org/abs/2107.13363>

We study competition among simultaneous heterogeneous contest designers in a general model that allows for a large space of contest design. Contestants choose in which contest to participate, and the goal of each contest designer is to maximize the contestants’ sum of efforts exerted in her contest. Our main result shows that, with symmetric contestants, optimal contests in the monopolistic setting (i.e., those that maximize the sum of efforts in a model with a single contest designer) form Pareto-optimal equilibria when contest designers compete. Under a natural assumption, monopolistic optimal contests are in fact dominant in the competitive case, and the equilibria that they form are unique. In many natural cases, they also maximize social welfare.

3.8 Dynamics in Network Creation and Residential Segregation

Pascal Lenzner (Hasso-Plattner-Institut, Universität Potsdam, DE)

License © Creative Commons BY 4.0 International license
© Pascal Lenzner

I will focus on dynamics in two different game-theoretic models, one focusing on network creation and the other one modelling location choice in a residential area.

Network creation games are motivated by the observation that the structure of many real-world networks like the Internet or (online) social networks is the outcome of a complex interaction of selfish agents. These settings can be modeled as strategic games and methods from Algorithmic Game Theory can be employed to rigorously analyze them. I will survey recent results on popular variants of network creation games with a focus on the most intriguing open problem: finding equilibria via suitable game dynamics.

Based on Schelling’s seminal model for residential segregation, recently game-theoretic variants of it have become popular. In these models agents with different types strategically select a location on a given graph that models the residential area. Each agent’s utility depends on the agent type distribution in its respective neighborhood. Given this, studying the game dynamics of such models provides insights into why large metropolitan areas are segregated along various dimensions like ethnicity or household income. I will survey recent work on these dynamic aspects.

3.9 Mobile Games: An Exciting New Domain for Behavioral Game Theory

Kevin Leyton-Brown (University of British Columbia – Vancouver, CA)

License © Creative Commons BY 4.0 International license
© Kevin Leyton-Brown

Joint work of Kevin Leyton-Brown, Taylor Lundy, Narun Raman, Hu Fu

Two and a half billion people play mobile games worldwide, spending \$180.3B USD in 2021 alone. More than half of this revenue comes from free-to-play mobile games. These games make money through carefully optimized monetization strategies based on optional in-game purchases. Mobile games also often leverage social dynamics to recruit new players and convert existing players into paying customers. Because player behavior in games, while thoughtful, often deviates from standard definitions of rationality, mobile games constitute a rich laboratory for research in behavioural game theory.

In our first work on this topic we looked at the phenomenon of “skips” in which players pay real money to avoid game content. We leverage existing tools of mechanism design to answer questions like “Why would a player ever value skips?” and “Can expensive skips be good for player welfare?”. The talk will focus particularly on open questions from our ongoing work, which investigates elaborating both the behavioral model of player engagement (drawing on recent work from Kleinberg et. al.); incorporating the effects that virality and social networks can have on a game’s success; and studying the pervasive monetization scheme in which players are given a handful of “lives” and are forced to wait for a timer to reset these are depleted (or to buy more lives).

3.10 On new variants of multiplicative weights update and mirror prox methods for 0-sum games

Vangelis Markakis (Athens University of Economics and Business, GR)

License © Creative Commons BY 4.0 International license
© Vangelis Markakis

Main reference Michail Fasoulakis, Evangelos Markakis, Yannis Pantazis, Constantinos Varsos: “Forward Looking Best-Response Multiplicative Weights Update Methods for Bilinear Zero-sum Games”, in Proc. of the Proceedings of The 25th International Conference on Artificial Intelligence and Statistics, Proceedings of Machine Learning Research, Vol. 151, pp. 11096–11117, PMLR, 2022.

URL <https://proceedings.mlr.press/v151/fasoulakis22a.html>

We follow up on a recent line of works that attempt to establish last-iterate convergence results for iterative first-order methods in min-max optimization. Our work focuses on extra gradient learning algorithms for finding Nash equilibria in bilinear zero-sum games, motivated by related questions in online learning. A typical drawback of several first-order methods, including the standard versions of gradient descent and multiplicative weight updates, is that they converge only in an average sense, but not w.r.t. their last iterate. We propose a new algorithm, that can be considered as a variant of the Mirror Prox method (Nemirovski 2004), using a large learning rate parameter for the intermediate gradient step and a small learning rate in the final step of each iteration. Although counter-intuitive at first sight due to the irrationally large intermediate rate, we prove that this method attains last-iterate convergence. Furthermore, we perform experimental comparisons against other recently proposed algorithms (such as the optimistic variant of the multiplicative weights update method, by Daskalakis and Panageas (2019)), and show that our algorithm has significant practical potential since it offers substantial gains in terms of accelerated convergence.

3.11 Byzantine Adversaries in Social Choice

Darya Melnyk (Aalto University, FI)

License  Creative Commons BY 4.0 International license
© Darya Melnyk

Main reference Darya Melnyk, Yuyi Wang, Roger Wattenhofer: “Byzantine Preferential Voting”, in Proc. of the Web and Internet Economics, pp. 327–340, Springer International Publishing, 2018.

URL https://doi.org/10.1007/978-3-030-04612-5_22

In distributed computing, robust communication algorithms that handle a wide variety of adversaries, including network failures, node crashes, and even unpredictable behavior of the network participants, are being developed. In social choice theory, on the other hand, the participants are usually assumed to be selfish or biased towards some opinion, and they only influence the network with this goal in mind. In this talk, I will show how worst-case behavior of participants, the so-called Byzantine behavior, can influence the outcome of different voting protocols. To this end, I will talk about different ways of measuring how much the Byzantine party is able to influence the final result. These results can be used to design voting protocols that better represent the opinion of truthful voters.

3.12 Predictably Irrational? Can behavioral decision making research help predict human choice? (Tutorial)


Ori Plonsky (Technion – Haifa, IL)

License  Creative Commons BY 4.0 International license
© Ori Plonsky

Humans are said to be predictably irrational, yet accurate predictions of human choice behavior remain a major challenge. Behavioral science research often demonstrates contradictory deviations from rational choice, and predicting the direction of deviation is not easy. In this talk, I will review some of the classical findings and the most robust behavioral tendencies in human choice behavior. I will then demonstrate when and how understanding and accounting for them can aid the development of state-of-the-art predictive models.

3.13 New Fairness notions for Resource Allocation

Nidhi Rathi (Aarhus University, DK)

License  Creative Commons BY 4.0 International license
© Nidhi Rathi

Joint work of Ioannis Caragiannis, Jugal Garg, Nidhi Rathi, Eklavya Sharma, Giovanna Varricchio

We consider the fundamental problem in resource allocation setting that entails fairly dividing a set of discrete goods among agents. The notion of envy-freeness up to any good (EFX) is the most compelling notion of fairness in this line of work. We say an allocation is EFX when every agent (weakly) prefers her own bundle than any other agent j 's bundle after removing her least positively-valued item from j 's bundle. Despite significant efforts over the past few years, existence of EFX allocations is not known even for additive valuations. Therefore, there has been a lot of work focused on finding relaxations and approximations of EFX. In this talk, I will propose two natural relaxations (one comparison-based and one threshold-based) of fairness notions that is inspired from EFX.

The first notion is that of epistemic EFX (EEFX) which extends the definition of epistemic envy-freeness from [ABC+18]. An allocation is said to be EEFX if for every agent i , it is possible to shuffle/redistribute the goods of the other agents such that agent i does not envy any other agent up to any good (in this new allocation). Interestingly, we show that EEFX allocations are always guaranteed to exist for additive valuations, and can be found in polynomial-time. Next, we introduce a threshold-based fairness criteria of minimum EFX value (MXS) with agent thresholds defined using EFX allocations. The MXS threshold for agent i is defined to be the minimum value she receives in any allocation where she does not envy any other agent up to any good. Finally, we prove that it is NP-hard to compute the MXS threshold of any agent in a given fair-division instance. It is relevant to note that, despite this hardness constraint, we prove that an MXS allocation can always be computed in polynomial time for instances with additive valuations.

3.14 Allocating Indivisible Goods to Strategic Agents: Pure Nash Equilibria and Fairness

Rebecca Reiffenhäuser (University of Amsterdam, NL)

License © Creative Commons BY 4.0 International license
© Rebecca Reiffenhäuser

Main reference Georgios Amanatidis, Georgios Birmpas, Federico Fusco, Philip Lazos, Stefano Leonardi, Rebecca Reiffenhäuser: “Allocating Indivisible Goods to Strategic Agents: Pure Nash Equilibria and Fairness”, in Proc. of the Web and Internet Economics – 17th International Conference, WINE 2021, Potsdam, Germany, December 14-17, 2021, Proceedings, Lecture Notes in Computer Science, Vol. 13112, pp. 149–166, Springer, 2021.

URL https://doi.org/10.1007/978-3-030-94676-0_9

When allocating goods to agents (with additive valuations), both fairness and incentive compatibility are central goals. Unfortunately, each of those is often hard to achieve on its own, and aiming for both in the same (money-free) routine quickly leads to strong impossibility results. This work suggests a way around those, by abandoning the initial goal for a notion of the next-best thing: we show that the very natural round-robin algorithm for additive valuations actually produces EF1-fair, “truthful equilibria” – even though agents might have misreported their values.

3.15 Asynchronous Opinion Dynamics in Social Networks

Daniel Schmand (Universität Bremen, DE)

License © Creative Commons BY 4.0 International license
© Daniel Schmand

Opinion spreading in a society decides the fate of elections, the success of products, and the impact of political or social movements. The model by Hegselmann and Krause is a well-known theoretical model to study such opinion formation processes in social networks. In contrast to many other theoretical models, it does not converge towards a situation where all agents agree on the same opinion. Instead, it assumes that people find an opinion reasonable if and only if it is close to their own. The system converges towards a stable situation where agents sharing the same opinion form a cluster, and agents in different clusters do not influence each other.

We focus on the social variant of the Hegselmann-Krause model where agents are connected by a social network and their opinions evolve in an iterative process. When activated, an agent adopts the average of the opinions of its neighbors having a similar opinion. By this, the set of influencing neighbors of an agent may change over time. To the best of our knowledge, social Hegselmann-Krause systems with asynchronous opinion updates have only been studied with the complete graph as social network. We show that such opinion dynamics with random agent activation are guaranteed to converge for any social network. We provide an upper bound of $O(n|E|^2(\varepsilon/\delta)^2)$ on the expected number of opinion updates until convergence, where $|E|$ is the number of edges of the social network. For the complete social network we show a bound of $O(n^3(n^2 + (\varepsilon/\delta)^2))$ that represents a major improvement over the previously best upper bound of $O(n^9(\varepsilon/\delta)^2)$. Our bounds are complemented by simulations that indicate asymptotically matching lower bounds.

3.16 Best of Both Worlds: Agents with Entitlements

Giovanna Varricchio (Goethe-Universität – Frankfurt am Main, DE)

License  Creative Commons BY 4.0 International license
© Giovanna Varricchio

Joint work of Martin Hoefer, Marco Schmalhofer, Giovanna Varricchio
Main reference Martin Hoefer, Marco Schmalhofer, Giovanna Varricchio: “Best of Both Worlds: Agents with Entitlements”, CoRR, Vol. abs/2209.03908, 2022.
URL <https://doi.org/10.48550/arXiv.2209.03908>

Fair division of indivisible goods is a central challenge in artificial intelligence. For many prominent fairness criteria including envy-freeness (EF) or proportionality (PROP), no allocations satisfying these criteria might exist. Two popular remedies to this problem are randomization or relaxation of fairness concepts. A timely research direction is to combine the advantages of both, commonly referred to as Best of Both Worlds (BoBW).

In the case of equally entitled agents and additive valuations, a lottery that is simultaneously ex-ante EF and ex-post envy-free up to one good (EF1) is known to exist. With our work, we broaden the picture by focusing on agents with different entitlements. Our main result is a lottery that is ex-ante weighted envy-free (WEF), as well as ex-post weighted proportional up to one good (WPROP1) and weighted transfer envy-free up to one good (WEF(1, 1)). In addition, we show that this result is tight – ex-ante WEF is incompatible with any stronger ex-post WEF relaxation.

We also try to extend positive results to more expressive valuation functions; unfortunately, our techniques partially apply in more general settings. Extending BoBW results is an interesting and challenging problem, and developing new tools is a fundamental next step in this direction.

3.17 Systemic risk in financial networks – a computational perspective

Carmine Ventre (King's College London, GB)

License © Creative Commons BY 4.0 International license
© Carmine Ventre

Joint work of Stavros Ioannidis, Bart de Keijzer, Carmine Ventre

We are given a network modelling assets and liabilities of the financial institutions in the system. We study the following basic question: Can we efficiently compute the exposure rate of each bank to defaults in the system?

The answer is yes if there are no financial derivatives (i.e., conditional obligations) in the network. When we introduce derivatives, specifically Credit Default Swaps (CDS), the problem is complete for the complexity class PPAD if one is content with "almost" (that is, weakly approximate) solutions that could grossly under- and over-estimate each bank's exposure.

What about solutions where the rate is precise to say 1%? We prove that computing these strong approximations up to any given precision is complete for the class FIXP, capturing hardness due to numerical aspects (in particular, the irrationality of the actual solution) in addition to the combinatorial issues modelled by PPAD.

We also study the relationship between the network structure and the (ir)rationality of the solution, the robustness of our findings to different payment rules of insolvent banks and the computational complexity of questions motivated by the needs of regulators. Overall, our results support a ban of the purely speculative naked CDS and uncover a connection between FIXP and efficient algorithms in the real computational model of Blum-Shub-Smale.

3.18 Does Computation Change Society? (Tutorial)

Roger Wattenhofer (ETH Zürich, CH)

License © Creative Commons BY 4.0 International license
© Roger Wattenhofer

In our Dagstuhl Seminar, we talk about better mechanisms to organize problems of society in a better (for instance fairer) way. We believe that by now computation should have had a huge impact on society, governments, and markets. However, many government mechanisms are still organized the same way as 100 years ago, pretty much ignoring all technical progress. In this talk, we discuss a few examples of computation in society. In the first part, we discuss systems that can help making voting decisions, or the benefits of electronic voting systems. We even fantasize about having a democracy without the need of politicians (made scalable by artificial intelligence). In the second part of the talk, we discuss some aspects of markets, in particular decentralized exchanges.

3.19 Seniorities and Minimal Clearing in Financial Network Games

Lisa Wilhelmi (Goethe-Universität – Frankfurt am Main, DE)

License © Creative Commons BY 4.0 International license
© Lisa Wilhelmi

Joint work of Martin Hoefer, Lisa Wilhelmi

Main reference Martin Hoefer, Lisa Wilhelmi: “Seniorities and Minimal Clearing in Financial Network Games”, in Proc. of the Algorithmic Game Theory – 15th International Symposium, SAGT 2022, Colchester, UK, September 12-15, 2022, Proceedings, Lecture Notes in Computer Science, Vol. 13584, pp. 187–204, Springer, 2022.

URL https://doi.org/10.1007/978-3-031-15714-1_11

Financial network games model payment incentives in the context of networked liabilities. In this paper, we advance the understanding of incentives in financial networks in two important directions: minimal clearing (arising, e.g., as a result of sequential execution of payments) and seniorities (i.e., priorities over debt contracts).

We distinguish between priorities that are chosen endogenously or exogenously. For endogenous priorities and standard (maximal) clearing, the games exhibit a coalitional form of weak acyclicity. A strong equilibrium exists and can be reached after a polynomial number of deviations. Moreover, there is a strong equilibrium that is optimal for a wide variety of social welfare functions. In contrast, for minimal clearing there are games in which no optimal strategy profile exists, even for standard utilitarian social welfare. Perhaps surprisingly, a strong equilibrium still exists and, for a wide range of strategies, can be reached after a polynomial number of deviations. In contrast, for exogenous priorities, equilibria can be absent and equilibrium existence is NP-hard to decide, for both minimal and maximal clearing.

3.20 Incentive-Compatible Forecasting Competitions

Jens Witkowski (Frankfurt School of Finance & Management, DE)

License © Creative Commons BY 4.0 International license
© Jens Witkowski

Joint work of Jens Witkowski, Rupert Freeman, Jennifer Wortman Vaughan, David Pennock, Andreas Krause

We initiate the study of incentive-compatible forecasting competitions in which multiple forecasters make predictions about one or more events and compete for a single prize. We have two objectives: (1) to incentivize forecasters to report truthfully and (2) to award the prize to the most accurate forecaster. Proper scoring rules incentivize truthful reporting if all forecasters are paid according to their scores. However, incentives become distorted if only the best-scoring forecaster wins a prize, since forecasters can often increase their probability of having the highest score by reporting more extreme beliefs. In this paper, we introduce two novel forecasting competition mechanisms. Our first mechanism is incentive compatible and guaranteed to select the most accurate forecaster with probability higher than any other forecaster. Moreover, we show that in the standard single-event, two-forecaster setting and under mild technical conditions, no other incentive-compatible mechanism selects the most accurate forecaster with higher probability. Our second mechanism is incentive compatible when forecasters’ beliefs are such that information about one event does not lead to belief updates on other events, and it selects the best forecaster with probability approaching 1 as the number of events grows. Our notion of incentive compatibility is more general than previous definitions of dominant strategy incentive compatibility in that it allows for reports to be correlated with the event outcomes. Moreover, our mechanisms are easy to implement and can be generalized to the related problems of outputting a ranking over forecasters and hiring a forecaster with high accuracy on future events.

3.21 A Simple Vision for Fair Division

Yair Zick (University of Massachusetts – Amherst, US)

License © Creative Commons BY 4.0 International license
© Yair Zick

Main reference Vignesh Viswanathan, Yair Zick: “A General Framework for Fair Allocation with Matroid Rank Valuations”, CoRR, Vol. abs/2208.07311, 2022.

URL <https://doi.org/10.48550/arXiv.2208.07311>

We study the problem of fairly allocating a set of indivisible goods among agents with matroid rank valuations. We present a simple framework that efficiently computes any fairness objective that satisfies some mild assumptions. Along with maximizing a fairness objective, the framework is guaranteed to run in polynomial time, maximize utilitarian social welfare and ensure strategyproofness. We show how our framework can be used to achieve four different fairness objectives: (a) Prioritized Lorenz dominance, (b) Maxmin fairness, (c) Weighted leximin, and (d) Max weighted Nash welfare. In particular, our framework provides the first polynomial time algorithms to compute weighted leximin and max weighted Nash welfare allocations for matroid rank valuations.

4 Open problems

4.1 Condorcet outcome for m independent binary issues

Andrei Constantinescu (ETH Zürich, CH)

License © Creative Commons BY 4.0 International license
© Andrei Constantinescu

An assembly of n voters wants to decide on m independent binary issues. The preferences of each voter and potential outcomes of the decision-making process are represented by m -bit vectors. In a direct race between two potential outcomes, a voter will vote for the outcome closer in Hamming distance to their vector, abstaining if indifferent.

Problem: Find if there is a Condorcet outcome; i.e. one not defeated by any other outcome in a direct race.

4.2 Interval cover for spatial crowdsourcing – Improving approximation

Vangelis Markakis (Athens University of Economics and Business, GR)

License © Creative Commons BY 4.0 International license
© Vangelis Markakis

Consider a set of tasks, say $\{1, \dots, m\}$, that are ordered in a line, and a set of available workers $\{1, \dots, n\}$. Each worker is able to work on a subinterval T_i of tasks for a cost c_i . A worker, if selected, can contribute with a value q_i to the tasks in the interval T_i , and each task j has a demand of d_j . The goal is to find an interval cover that completes every task and is of minimum cost. The optimization problem is known to be NP-hard but not APX-hard. Furthermore, the best-known approximation is 3.


Question 1: What’s the best approximation for the minimization problem?

Consider the strategic variant of the problem where costs are private information of workers. Denoted by Δ the maximum number of workers that can contribute to a task, the best-known approximation for a strategyproof mechanism is of Δ .

Question 2: What's the best truthful approximation for the minimization problem?

4.3 Complexity of Public Goods Games on Graph – Nash equilibrium existence (Solved)

Noam Nisan (*The Hebrew University of Jerusalem, IL*)

License  Creative Commons BY 4.0 International license

© Noam Nisan

Main reference Matan Gilboa, Noam Nisan: “Complexity of Public Goods Games on Graphs”, CoRR, Vol. abs/2207.04238, 2022.

URL <https://doi.org/10.48550/arXiv.2207.04238>

This is a specific problem that was left open in the recent paper “Complexity of Public Goods Games on Graphs” by M. Gilboa and N. Nisan, SAGT 2022.

The game is played on a graph $G = (V, E)$ where each node is an agent that needs to choose among two actions $\{1, 0\}$. The action “1” is a best reply of a node if and only if at most two of its neighbours in the graph play 1. A pure Nash equilibrium of the game is an assignment $a : V \rightarrow \{0, 1\}$ such that for every vertex v we have:

$$a(v) = 1 \text{ iff } |\{u \mid (v, u) \in E \text{ and } a(u) = 1\}| \leq 2.$$

Question: Is it true that for every G there exists a pure Nash Equilibrium? Can you find one in polynomial time?

Solution: During the Dagstuhl Seminar, M. Klimm came up with an elegant solution by showing this game is equivalent to a congestion game, which implies that best-response dynamics converge to a pure Nash equilibrium in polynomial time. This solution was subsequently developed by M. Klimm and M.J. Stahlberg into their recent manuscript “Complexity of equilibria in binary public goods games on undirected graphs” (see <https://arxiv.org/pdf/2301.11849.pdf>).

Worth of mentioning is the independent and concurrent work “A Characterization of Complexity in Public Goods Games” by M. Gilboa (see <https://arxiv.org/pdf/2301.11580.pdf>).

4.4 Optimizing over Serial Dictatorships – Lower bound for optimal matching

Nidhi Rathi (*Aarhus University, DK*)

License  Creative Commons BY 4.0 International license

© Nidhi Rathi

Main reference Ioannis Caragiannis, Nidhi Rathi: “Optimizing over Serial Dictatorships”, CoRR, Vol. abs/2202.08097, 2022.

URL <https://arxiv.org/abs/2202.08097>

Consider a bipartite graph with n agents on one side and items on another. We access the weights via making following queries. On a query of the form (agent i , an ordered sequence S of agents), the oracle outputs the highest-valued item available for agent i , when she gets

to pick after the sequence of agents S . Our recent work proves that there always exists an action sequence that results into a maximum weight matching. We also prove that we can find an action sequence with $O(n^2)$ queries that produces a 2-approximate matching.

Question: Can we improve the lower bound on the number of queries or the approximation factor to the optimal matching?

4.5 Altruistic Hedonic Games (Short talk)

Jörg Rothe (*Heinrich-Heine-Universität Düsseldorf, DE*)

License © Creative Commons BY 4.0 International license
© Jörg Rothe

Hedonic games are coalition formation games in which players have preferences over the coalitions they can join. While previous models of representing hedonic games were based upon selfish players only, in a JAIR 2022 paper we have proposed a novel model for them that takes into account not only the players' own preferences but also their friends' preferences. In this talk, we briefly introduce our model, survey some known results, and outline open questions.

4.6 Tractability frontier of justice criteria under submodular valuations

Yair Zick (*University of Massachusetts – Amherst, US*)

License © Creative Commons BY 4.0 International license
© Yair Zick

When agents have binary submodular valuation functions over items, we can efficiently compute allocations that satisfy several justice criteria (leximin, EF-X, max. Nash welfare, (approximate) MMS, and more).

What happens when we move beyond binary valuations?

Recently, we were able to prove that when agents have submodular valuations with bivalued marginal gains (say, a and b), it is possible to still efficiently compute allocations that satisfy weaker notions of justice, when a divides b .

It is also known that for general submodular valuations, it is impossible to efficiently compute even relatively mild justice criteria (e.g. finding an allocation that maximizes the minimum agent welfare, also known as the Santa Claus problem), let alone stronger ones like leximin allocations.

Where is the tractability frontier? Can we efficiently compute just outcomes with trinary valuations (under some assumptions)? What happens when we assume that all marginal gains follow a certain distribution? Understanding these domains will allow us to move towards better algorithmic frameworks for fair allocation.

Participants

- Yakov Babichenko
Technion – Haifa, IL
- Gerdus Benadè
Boston University, US
- Ioannis Caragiannis
Aarhus University, DK
- Giorgos Christodoulou
Aristotle University of
Thessaloniki, GR
- Andrei Constantinescu
ETH Zürich, CH
- Michal Feldman
Tel Aviv University, IL
- Tobias Harks
Universität Passau, DE
- Martin Hofer
Goethe-Universität –
Frankfurt am Main, DE
- Max Klimm
TU Berlin, DE
- Ron Lavi
University of Bath, GB
- Pascal Lenzner
Hasso-Plattner-Institut,
Universität Potsdam, DE
- Kevin Leyton-Brown
University of British Columbia –
Vancouver, CA
- Vangelis Markakis
Athens University of Economics
and Business, GR
- Darya Melnyk
Aalto University, FI
- Noam Nisan
The Hebrew University of
Jerusalem, IL
- Sigal Oren
Ben Gurion University –
Beer Sheva, IL
- Ori Plonsky
Technion – Haifa, IL
- Maria Polukarov
King's College London, GB
- Nidhi Rathi
Aarhus University, DK
- Rebecca Reiffenhäuser
University of Amsterdam, NL
- Jörg Rothe
Heinrich-Heine-Universität
Düsseldorf, DE
- Daniel Schmand
Universität Bremen, DE
- Giovanna Varricchio
Goethe-Universität –
Frankfurt am Main, DE
- Carmine Ventre
King's College London, GB
- Roger Wattenhofer
ETH Zürich, CH
- Lisa Wilhelmi
Goethe-Universität – Frankfurt
am Main, DE
- Jens Witkowski
Frankfurt School of Finance &
Management, DE
- Yair Zick
University of Massachusetts –
Amherst, US



Dynamic Graph Algorithms

Aaron Bernstein^{*1}, Shiri Chechik^{*2}, Sebastian Forster^{*3},
Tsvi Kopelowitz^{*4}, Yasamin Nazari^{†5}, and Nicole Wein^{†6}

- 1 Rutgers University – New Brunswick, US. aaron.bernstein@rutgers.edu
- 2 Tel Aviv University, IL. shiri.chechik@gmail.com
- 3 Universität Salzburg, AT. forster@cs.sbg.ac.at
- 4 Bar-Ilan University – Ramat Gan, IL. kopelot@gmail.com
- 5 Universität Salzburg, AT. ynazari@cs.sbg.ac.at
- 6 Rutgers University – Piscataway, US. nwein@mit.edu

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 22461 “Dynamic Graph Algorithms”, which took place from November 13 to November 18, 2022.

The field of dynamic graph algorithms studies algorithms for processing graphs that are changing over time. Formally, the goal is to process an interleaved sequence of update and query operations, where an update operation changes the input graph (e.g. inserts/deletes an edge), while the query operation is problem-specific and asks for some information about the current graph – for example, an s - t path, or a minimum spanning tree. The field has evolved rapidly over the past decade, and this Dagstuhl Seminar brought together leading researchers in dynamic algorithms and related areas of graph algorithms.

Seminar November 13–18, 2022 – <http://www.dagstuhl.de/22461>

2012 ACM Subject Classification Theory of computation → Dynamic graph algorithms

Keywords and phrases dynamic graphs, graph algorithms

Digital Object Identifier 10.4230/DagRep.12.11.45

1 Executive Summary

Sebastian Forster (Universität Salzburg, AT)

Aaron Bernstein (Rutgers University – New Brunswick, US)

Shiri Chechik (Tel Aviv University, IL)

Tsvi Kopelowitz (Bar-Ilan University – Ramat Gan, IL)

License  Creative Commons BY 4.0 International license
© Sebastian Forster, Aaron Bernstein, Shiri Chechik, and Tsvi Kopelowitz

The field of dynamic graph algorithms has evolved rapidly over the past decade. New techniques, new problems, new lower bounds, and new approaches have yielded an extremely fruitful research environment. This seminar provided a venue for the community to establish the main challenges that remain and to actively shape the direction of the field going forward.

The seminar brought together the leading researchers and “rising stars” of the field as well as experts in “neighboring” areas such as distributed algorithms, parallel algorithms, streaming algorithms, online algorithms, approximation algorithms, data structures, fine-grained and parameterized complexity, and optimization. Many participants were also actively researching algorithms engineering for dynamic graph problems, which added interesting perspectives on the prevalent theory-practice gap and fundamental methodological challenges.

* Editor / Organizer

† Editorial Assistant / Collector



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

Dynamic Graph Algorithms, *Dagstuhl Reports*, Vol. 12, Issue 11, pp. 45–65

Editors: Aaron Bernstein, Shiri Chechik, Sebastian Forster, Tsvi Kopelowitz, Yasamin Nazari, and Nicole Wein



Dagstuhl Reports

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Several participants gave talks that were highlighting “cutting-edge” advances in the field, including results that very recently appeared in top theory venues such as STOC, FOCS and SODA (and in some cases won the best paper award). Some participants explored connections to other related areas in algorithms research such as distributed and streaming algorithms or differential privacy. Many of the talks also included suggestions on future directions and highlighted the main challenges in the area.

In two open problem sessions, the participants explicitly identified several central open problems that continue to resist progress. We hope that the resulting list of open problems will be a valuable resource for future research in this field.

During and after the sessions, attendees participated in vibrant discussions. Such interactions enhanced the overall experience and made a clear distinction to a traditional conference-like format. In addition to the research activities, there were many social activities (such as board games, poker, music night, hiking, and ping pong) that made the workshop also a great networking opportunity, in particular for the relatively large fraction of participants who were first-time visitors to Schloss Dagstuhl.

Acknowledgments

The organizers would like to thank Yasamin Nazari and Nicole Wein for helping edit this report, Monika Henzinger for helping develop the concept of this seminar, and the Dagstuhl team for their first-rate support.

2 Table of Contents

Executive Summary

Sebastian Forster, Aaron Bernstein, Shiri Chechik, and Tsvi Kopelowitz 45

Overview of Talks

Dynamic Distributed Subgraph Finding <i>Keren Censor-Hillel</i>	49
Stronger 3-SUM Lower Bounds for Approximate Distance Oracles via Additive Combinatorics <i>Nick Fischer</i>	49
Deterministic Incremental APSP with Polylogarithmic Update Time and Stretch <i>Sebastian Forster</i>	49
Incremental Approximate Maximum Flow in $m^{1/2+o(1)}$ update time <i>Gramoz Goranci</i>	50
Fully Dynamic Graph Algorithms in Practice: (Some) Lessons Learned <i>Kathrin Hanauer</i>	50
Strongly polynomial dynamic algorithms for minimum-weight cycle and related problems <i>Adam Karczmarz</i>	51
Dynamic Algorithms for Packing-Covering LPs via Multiplicative Weight Updates <i>Peter Kiss</i>	51
Balanced Allocations: The Heavy Case With Deletions <i>William Kuszmaul</i>	52
Optimal Decremental Connectivity in Non-Sparse Graphs <i>Jakub Łącki</i>	52
Online Routing and Network Design with Predictions <i>Nicole Megow</i>	53
Deterministic Fully Dynamic Distance Approximation <i>Yasamin Nazari</i>	53
Scalable dynamic graph processing with low latency: insights and challenges <i>Nikos Parotsidis</i>	54
A Blackbox Reduction for Adaptive Adversaries using Differential Privacy <i>Thatchaphol Saranurak</i>	54
Recent Results in Engineering Dynamic Graph Algorithms <i>Christian Schulz</i>	54
Queuing Safely for Elevator Systems Amidst a Pandemic <i>Clifford Stein</i>	55
Dynamic Graph Sketching: To Infinity And Beyond <i>David Tench</i>	55
Dynamic Matching with Better-than-2 Approximation in Polylogarithmic Update Time <i>David Wajc</i>	56

Dynamic Distance Oracles in Planar Graphs <i>Oren Weimann</i>	56
Optimal resizable arrays <i>Uri Zwick</i>	56
Open problems	
Reducing weighted matching to unweighted matching <i>Aaron Bernstein</i>	57
Communication Complexity of Max-Flow <i>Joakim Blikstad</i>	58
Dynamic Complexity of Low-Stretch Spanning Trees <i>Gramoz Goranci</i>	59
Polylog query time for a dynamic all-pairs problem in plane directed graphs <i>Adam Karczmarz</i>	60
Breaking CountMin Sketches Inside a Greedy Outer Loop <i>Richard Peng</i>	61
Dynamic Maximal Matching <i>Shay Solomon</i>	62
Dynamic Derandomization <i>David Wajc</i>	62
Participants	65

3 Overview of Talks

3.1 Dynamic Distributed Subgraph Finding

Keren Censor-Hillel (Technion – Haifa, IL)

License © Creative Commons BY 4.0 International license
© Keren Censor-Hillel

This talk is a discussion on exciting recent progress in distributed subgraph finding (static and dynamic) and describing some of the many intriguing open questions.

3.2 Stronger 3-SUM Lower Bounds for Approximate Distance Oracles via Additive Combinatorics

Nick Fischer (MPI für Informatik – Saarbrücken, DE)

License © Creative Commons BY 4.0 International license
© Nick Fischer

Joint work of Amir Abboud, Karl Bringmann, Nick Fischer

Main reference Amir Abboud, Karl Bringmann, Nick Fischer: “Stronger 3-SUM Lower Bounds for Approximate Distance Oracles via Additive Combinatorics”, CoRR, Vol. abs/2211.07058, 2022.

URL <https://doi.org/10.48550/arXiv.2211.07058>

In this work we prove conditional lower bounds against approximate distance oracles in static and dynamic settings. The seminal Thorup-Zwick distance oracles achieve stretch $2k \pm O(1)$ after preprocessing a graph in $O(mn^{1/k})$ time. For the same stretch, and assuming the query time is $n^{o(1)}$, Abboud, Bringmann, Khoury and Zamir (STOC ’22) proved an $\Omega(m^{1+\frac{1}{12.7552 \cdot k}})$ lower bound on the preprocessing time; we improve it to $\Omega(m^{1+1/2k})$ which is only a factor 2 away from the upper bound. Additionally, we obtain tight bounds for stretch $3 - \epsilon$ and higher lower bounds for dynamic shortest paths.

3.3 Deterministic Incremental APSP with Polylogarithmic Update Time and Stretch

Sebastian Forster (Universität Salzburg, AT)

License © Creative Commons BY 4.0 International license
© Sebastian Forster

Joint work of Sebastian Forster, Yasamin Nazari, Maximilian Probst Gutenberg

Main reference Sebastian Forster, Yasamin Nazari, Maximilian Probst Gutenberg: “Deterministic Incremental APSP with Polylogarithmic Update Time and Stretch”, CoRR, Vol. abs/2211.04217, 2022.

URL <https://doi.org/10.48550/arXiv.2211.04217>

We provide the first deterministic data structure that given a weighted undirected graph undergoing edge insertions, processes each update with polylogarithmic amortized update time and answers queries for the distance between any pair of vertices in the current graph with a polylogarithmic approximation in $O(\log \log n)$ time.

Prior to this work, no data structure was known for partially dynamic graphs, i.e., graphs undergoing either edge insertions or deletions, with less than $n^{o(1)}$ update time except for dense graphs, even when allowing randomization against oblivious adversaries or considering only single-source distances.

3.4 Incremental Approximate Maximum Flow in $m^{1/2+o(1)}$ update time

Gramoz Goranci (ETH Zürich, CH)

License  Creative Commons BY 4.0 International license
 Gramoz Goranci

Joint work of Gramoz Goranci, Monika Henzinger



Main reference Gramoz Goranci, Monika Henzinger: “Incremental Approximate Maximum Flow in $m^{1/2+o(1)}$ update time”, CoRR, Vol. abs/2211.09606, 2022.

URL <https://doi.org/10.48550/arXiv.2211.09606>

We show a $(1 + \epsilon)$ -approximation algorithm for maintaining maximum s - t flow under m edge insertions in $m^{1/2+o(1)}\epsilon^{-1/2}$ amortized update time for directed, unweighted graphs. This constitutes the first sublinear dynamic maximum flow algorithm in general sparse graphs with arbitrarily good approximation guarantee.

3.5 Fully Dynamic Graph Algorithms in Practice: (Some) Lessons Learned

Kathrin Hanauer (Universität Wien, AT)

License  Creative Commons BY 4.0 International license
 Kathrin Hanauer

Joint work of Kathrin Hanauer, Monika Henzinger, Christian Schulz, Leonhard Paul Sidl

Main reference Kathrin Hanauer, Monika Henzinger, Christian Schulz: “Fully Dynamic Single-Source Reachability in Practice: An Experimental Study”, in Proc. of the Symposium on Algorithm Engineering and Experiments, ALENEX 2020, Salt Lake City, UT, USA, January 5-6, 2020, pp. 106–119, SIAM, 2020.

URL <https://doi.org/10.1137/1.9781611976007.9>

Fully dynamic graph algorithms have received growing attention in experimental work recently, though still by far not as much as they have in theory. In this talk, we will consider different approaches to maintain reachability information in a graph as it undergoes a series of insertions and deletions and analyze their behavior in practice on different types of fully dynamic instances.

The study of fully dynamic algorithms in practice is complicated by a lack of real-world, “real-dynamic” instances that are available publicly. In this talk will take a closer look at how the characteristics of an update sequence can influence the behavior of dynamic algorithms for reachability and subgraph counting and also review different approaches that are currently in use to overcome the shortage in fully dynamic instances in practice.

The talk concludes with a summary of lessons learned when engineering specifically dynamic graph algorithms.

3.6 Strongly polynomial dynamic algorithms for minimum-weight cycle and related problems

Adam Karczmarz (University of Warsaw, PL & IDEAS NCBR – Warsaw, PL)

License © Creative Commons BY 4.0 International license
© Adam Karczmarz

Main reference Adam Karczmarz: “Fully Dynamic Algorithms for Minimum Weight Cycle and Related Problems”, in Proc. of the 48th International Colloquium on Automata, Languages, and Programming, ICALP 2021, July 12-16, 2021, Glasgow, Scotland (Virtual Conference), LIPIcs, Vol. 198, pp. 83:1–83:20, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

URL <https://doi.org/10.4230/LIPIcs.ICALP.2021.83>

A relatively small portion of the known dynamic algorithms for shortest paths and related problems have strongly polynomial update bounds. This means, roughly speaking, that in most cases the update bounds either do not hold for real-weighted graphs or depend on the magnitude of the graph’s weights. One notable exception is the fully dynamic APSP algorithm of Demetrescu and Italiano [J.ACM’04].

In this talk, we will consider maintaining negative/minimum-weight/minimum-mean cycles in dynamic real-weighted digraphs. The best-known static strongly polynomial algorithms for these classical problems run in $O(nm)$ time. For some of these problems, non-trivial strongly polynomial update bounds can be obtained. For others, we will try to identify some challenges.

3.7 Dynamic Algorithms for Packing-Covering LPs via Multiplicative Weight Updates

Peter Kiss (University of Warwick – Coventry, GB)

License © Creative Commons BY 4.0 International license
© Peter Kiss

Joint work of Sayan Bhattacharya, Peter Kiss, Thatchaphol Saranurak


Main reference Sayan Bhattacharya, Peter Kiss, Thatchaphol Saranurak: “Dynamic Algorithms for Packing-Covering LPs via Multiplicative Weight Updates”, CoRR, Vol. abs/2207.07519, 2022.

URL <https://doi.org/10.48550/arXiv.2207.07519>

In the dynamic linear program (LP) problem, we are given an LP undergoing updates and we need to maintain an approximately optimal solution. Recently, significant attention (e.g., [Gupta et al. STOC’17; Arar et al. ICALP’18, Wajc STOC’20]) has been devoted to the study of special cases of dynamic packing and covering LPs, such as the dynamic fractional matching and set cover problems. But until now, there is no non-trivial dynamic algorithm for general packing and covering LPs. In this work, we settle the complexity of dynamic packing and covering LPs, up to a polylogarithmic factor in update time. More precisely, in the partially dynamic setting (where updates can either only relax or only restrict the feasible region), we give near-optimal deterministic ϵ -approximation algorithms with polylogarithmic amortized update time. Then, we show that both partially dynamic updates and amortized update time are necessary; without any of these conditions, the trivial algorithm that recomputes the solution from scratch after every update is essentially the best possible, assuming SETH. To obtain our results, we initiate a systematic study of the multiplicative weights update (MWU) method in the dynamic setting. As by-products of our techniques, we also obtain the first online $(1 + \epsilon)$ -competitive algorithms for both covering and packing LPs with polylogarithmic recourse, and the first streaming algorithms for covering and packing LPs with linear space and polylogarithmic passes.

3.8 Balanced Allocations: The Heavy Case With Deletions

William Kuszmaul (MIT – Cambridge, US)

License  Creative Commons BY 4.0 International license
© William Kuszmaul

Joint work of Nikhil Bansal, William Kuszmaul

Main reference Nikhil Bansal, William Kuszmaul: “Balanced Allocations: The Heavily Loaded Case with Deletions”, CoRR, Vol. abs/2205.06558, 2022.


URL <https://doi.org/10.48550/arXiv.2205.06558>

In the 2-choice allocation problem, m balls are placed into n bins, and each ball must choose between two random bins $i, j \in [n]$ that it has been assigned to. It has been known for more than two decades, that if each ball follows the GREEDY strategy (i.e., always pick the less-full bin), then the maximum load will be $m/n + O(\log \log n)$ with high probability in n (and $m/n + O(\log m)$ with high probability in m). It has remained an open question whether the same bounds hold in the *dynamic* version of the same game, where balls are inserted/deleted with no more than m balls present at a time.

We show that, somewhat surprisingly, these bounds do not hold in the dynamic setting: already on 4 bins, there exists a sequence of insertions/deletions that cause the GREEDY strategy to incur a maximum load of $m/4 + \Omega(\sqrt{m})$ with probability $\Omega(1)$. This raises the question of whether *any* 2-choice allocation strategy can offer a strong bound in the dynamic setting. Our second result answers this question in the affirmative: we present a new strategy, called MODULATEDGREEDY, that guarantees a maximum load of $m/n + O(\log m)$, at any given moment, with high probability in m .

3.9 Optimal Decremental Connectivity in Non-Sparse Graphs

Jakub Łącki (Google – New York, US)

License  Creative Commons BY 4.0 International license
© Jakub Łącki

Joint work of Anders Aamand, Adam Karczmarz, Jakub Łącki, Nikos Parotsidis, Peter M. R. Rasmussen, Mikkel Thorup

Main reference Anders Aamand, Adam Karczmarz, Jakub Lacki, Nikos Parotsidis, Peter M. R. Rasmussen, Mikkel Thorup: “Optimal Decremental Connectivity in Non-Sparse Graphs”, CoRR, Vol. abs/2111.09376, 2021.

URL <https://arxiv.org/abs/2111.09376>

We show an algorithm for decremental maintenance of connected components and 2-edge connected components, which handles any sequence of edge deletions in $O(m + npolylog n)$ time and answers queries in constant time. This talk focuses on three ideas behind this result: a new sparse connectivity certificate, which can be updated dynamically, a new way of using the XOR-trick, which allows one detect small cuts, and a self-check technique, which allows us to obtain a Las Vegas randomized algorithm based on a Monte Carlo data structure.

3.10 Online Routing and Network Design with Predictions

Nicole Megow (*Universität Bremen, DE*)

License © Creative Commons BY 4.0 International license
© Nicole Megow

Joint work of Nicole Megow, Giulia Bernardini, Alexander Lindermayr, Alberto Marchetti-Spaccamela, Leen Stougie, Michelle Sweering

Online optimization refers to solving problems where an initially unknown input is revealed incrementally, and irrevocable decisions must be made not knowing future requests. The assumption of not having any prior knowledge about future requests seems overly pessimistic. Given the success of machine-learning methods and data-driven applications, one may expect to have access to predictions about future requests. However, simply trusting them might lead to very poor solutions as these predictions come with no quality guarantee. In this talk we present recent developments in the young line of research that integrates such error-prone predictions into algorithm design to break through worst case barriers. We discuss algorithmic challenges with a focus on online routing and network design and present algorithms with performance guarantees depending on a novel error metric.

3.11 Deterministic Fully Dynamic Distance Approximation

Yasamin Nazari (*Universität Salzburg, AT*)

License © Creative Commons BY 4.0 International license
© Yasamin Nazari

Joint work of Jan van den Brand, Sebastian Forster, Yasamin Nazari

Main reference Jan van den Brand, Sebastian Forster, Yasamin Nazari: “Fast Deterministic Fully Dynamic Distance Approximation”, CoRR, Vol. abs/2111.03361, 2021.

URL <https://arxiv.org/abs/2111.03361>

The first part of the talk focuses on our deterministic fully dynamic algorithms for computing approximate distances in a graph. Specifically, we are given an unweighted and undirected graph $G = (V, E)$ undergoing edge insertions and deletions, and a parameter $0 < \epsilon \leq 1$, and our goal is to maintain $(1 + \epsilon)$ -approximate distances between a single pair (st distance), a single source to all nodes (SSSP), or all pairs (APSP). We discuss how combinatorial tools such as emulators can be combined with algebraic data structures to obtain deterministic algorithms with improved worst-case guarantees for these problems.

The second part of the talk focuses on future directions for obtaining improved fully dynamic algorithms for weighted or directed graphs. We explore possible candidate combinatorial structures that could be used and the challenges in maintaining them in the fully dynamic settings.

3.12 Scalable dynamic graph processing with low latency: insights and challenges

Nikos Parotsidis (Google Research – Zürich, CH)

License © Creative Commons BY 4.0 International license
© Nikos Parotsidis

In this talk we discuss insights from the development of a scalable system for processing dynamic graph algorithms with low latency inside Google. We discuss applications, requirements, and challenges that arise in such a real-world system. The three challenges that we discuss are 1) how to maintain a solution that does not change very drastically during the execution of the algorithm, 2) how to process a graph in a distributed and dynamic fashion; which is mandated by the scale of the data, and 3) how to process a large volume of concurrent updates, each within low latency. These challenges naturally lead us in exploring new models (and evaluating the suitability of existing models) for tackling them.

3.13 A Blackbox Reduction for Adaptive Adversaries using Differential Privacy

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

License © Creative Commons BY 4.0 International license
© Thatchaphol Saranurak

Main reference Amos Beimel, Haim Kaplan, Yishay Mansour, Kobbi Nissim, Thatchaphol Saranurak, Uri Stemmer: “Dynamic Algorithms Against an Adaptive Adversary: Generic Constructions and Lower Bounds”, CoRR, Vol. abs/2111.03980, 2021.

URL <https://arxiv.org/abs/2111.03980>

This talk is a tutorial on how to use differential privacy to obtain a black-box reduction that can transform any dynamic algorithm for any estimation problem that works against an oblivious adversary to another algorithm against an adaptive adversary.

3.14 Recent Results in Engineering Dynamic Graph Algorithms

Christian Schulz (Universität Heidelberg, DE)

License © Creative Commons BY 4.0 International license
© Christian Schulz

In recent years, significant advances have been made in the design and analysis of fully dynamic algorithms. However, these theoretical results have received very little attention from the practical perspective. Few of the algorithms are implemented and tested on real datasets, and their practical potential is far from understood. In this talk, we give a brief overview of results in engineering dynamic graph algorithms that we achieved recently. To this end, we give a high level overview of dynamic algorithms and their performance for (hyper) graph (b -)matching, independent sets, edge-orientation, reachability as well as k -center clustering and minimum cuts.

3.15 Queuing Safely for Elevator Systems Amidst a Pandemic

Clifford Stein (Columbia University, US)

License © Creative Commons BY 4.0 International license
© Clifford Stein

Joint work of Sai Mali Ananthanarayanan, Charles C. Branas, Adam Elmachtoub, Clifford Stein, Yeqing Zhou

Main reference Sai Mali Ananthanarayanan, Charles C. Branas, Adam Elmachtoub, Clifford Stein, Yeqing Zhou: “Queuing Safely for Elevator Systems Amidst a Pandemic” (December 21, 2020). Production and Operations Management, 00 1– 18. Available at SSRN.

URL <https://doi.org/10.1111/poms.13686>

The requirement of social distancing during the COVID-19 pandemic has presented significant challenges for high-rise buildings, which heavily rely on elevators for vertical transportation. In particular, the need for social distancing has reduced elevator capacity typically by at least two-thirds or as much as over 90% the normal amount. This reduction is a serious concern, as reduced elevator capacities cause large queues to build up in lobbies, which makes social distancing difficult and results in large wait times. The objective of this study is to safely manage the elevator queues by proposing simple, technology-free interventions that drastically reduce the waiting time and length of lobby queues. We use mathematical modeling, epidemiological expertise, and simulation to design and evaluate our interventions. The key idea is to explicitly or implicitly group passengers that are going to the same floor into the same elevator as much as possible. In the Cohorting intervention, we attempt to find passengers going to the same floor as the first person in the queue. In the Queue Splitting intervention, we create a different queue for different groups of floors. Based on simulation and analytical findings, Cohorting and Queue Splitting can significantly reduce queue length and wait time, while also maintaining safety from viral transmission in otherwise crowded elevators, building lobbies, and entrances. These interventions are generally accessible for many buildings since they do not require programming the elevators, and rely on only using signage and/or a queue manager to guide passengers.

3.16 Dynamic Graph Sketching: To Infinity And Beyond

David Tench (Rutgers University – Piscataway, US)

License © Creative Commons BY 4.0 International license
© David Tench

Joint work of David Tench, Evan West, Victor Zhang, Michael A. Bender, Abiyaz Chowdhury, J. Ahmed Deltas, Martin Farach-Colton, Tyler Seip, Kenny Zhang

Main reference David Tench, Evan West, Victor Zhang, Michael A. Bender, Abiyaz Chowdhury, J. Ahmed Deltas, Martin Farach-Colton, Tyler Seip, Kenny Zhang: “GraphZeppelin: Storage-Friendly Sketching for Connected Components on Dynamic Graph Streams”, in Proc. of the SIGMOD ’22: International Conference on Management of Data, Philadelphia, PA, USA, June 12 – 17, 2022, pp. 325–339, ACM, 2022.

URL <https://doi.org/10.1145/3514221.3526146>

Existing graph stream processing systems must store the graph explicitly in RAM which limits the scale of graphs they can process. The graph semi-streaming literature offers algorithms which avoid this limitation via linear sketching data structures that use small (sublinear) space, but these algorithms have not seen use in practice to date. This talk explores what is needed to make graph sketching algorithms practically useful, and as a case study present a sketching algorithm for connected components and a corresponding high-performance implementation. Finally, we give an overview of the many open problems in this area, focusing on improving query performance of graph sketching algorithms.

3.17 Dynamic Matching with Better-than-2 Approximation in Polylogarithmic Update Time

David Wajc (Stanford University, US)

License  Creative Commons BY 4.0 International license
© David Wajc

Joint work of Sayan Bhattacharya, Peter Kiss, Thatchaphol Saranurak, David Wajc
Main reference Sayan Bhattacharya, Peter Kiss, Thatchaphol Saranurak, David Wajc: “Dynamic Matching with Better-than-2 Approximation in Polylogarithmic Update Time”, CoRR, Vol. abs/2207.07438, 2022.
URL <https://doi.org/10.48550/arXiv.2207.07438>

We present dynamic algorithms with polylog update time for the value version of the dynamic matching problem with approximation ratio strictly better than 2. Specifically, we obtain a $1 + 1/\sqrt{2} + \epsilon \approx 1.707 + \epsilon$ approximation in bipartite graphs and a $1.973 + \epsilon$ approximation in general graphs.

3.18 Dynamic Distance Oracles in Planar Graphs


Oren Weimann (University of Haifa, IL)

License  Creative Commons BY 4.0 International license
© Oren Weimann

A distance oracle is a data structure for answering distance queries on a graph. While on general graphs efficient distance oracles must settle for approximate answers, on planar graphs recent progress has led to exact oracles with almost linear space and polylogarithmic query time (i.e. almost optimal). However, in the dynamic setting (when the underlying graph is subject to updates) there has been no significant progress in recent years. The state of the art is an exact oracle from more than 20 years ago that given a planar graph supports both updates and queries in $\tilde{O}(n^{2/3})$ time. On the lower-bound side, conditioned on the APSP hypothesis, in any dynamic exact distance oracle (in fact, even in the offline setting) either the update or the query must take $\Omega(n^{1/2})$ time, leaving an intriguing gap. For approximate distances, the currently fastest oracle requires $\tilde{O}(n^{1/2})$ time for both updates and queries (and there is no known lower bound), and in the offline setting there is an almost optimal solution with polylogarithmic time for both updates and queries. In this talk we will describe these upper and lower bounds (for exact distances), the tight connections in planar graphs between distance oracles and maximum-flow (or minimum-cut) oracles, concrete open problems, and possible directions for solving them.

3.19 Optimal resizable arrays

Uri Zwick (Tel Aviv University, IL)

License  Creative Commons BY 4.0 International license
© Uri Zwick

Joint work of Robert E. Tarjan, Uri Zwick
Main reference Robert E. Tarjan, Uri Zwick: “Optimal resizable arrays”, CoRR, Vol. abs/2211.11009, 2022.
URL <https://doi.org/10.48550/arXiv.2211.11009>

A *resizable array* is an array that can *grow* and *shrink* by the addition or removal of items from its end, or both its ends, while still supporting constant-time *access* to each item stored in the array given its *index*. Since the size of an array, i.e., the number of items

in it, varies over time, space-efficient maintenance of a resizable array requires dynamic memory management. A standard doubling technique allows the maintenance of an array of size N using only $O(N)$ space, with $O(1)$ amortized time, or even $O(1)$ worst-case time, per operation. Sitarski and Brodnik et al. describe much better solutions that maintain a resizable array of size N using only $N + O(\sqrt{N})$ space, still with $O(1)$ time per operation. Brodnik et al. give a simple proof that this is best possible.

We distinguish between the space needed for *storing* a resizable array, and accessing its items, and the *temporary* space that may be needed while growing or shrinking the array. For every integer $r \geq 2$, we show that $N + O(N^{1/r})$ space is sufficient for storing and accessing an array of size N , if $N + O(N^{1-1/r})$ space can be used briefly during grow and shrink operations. Accessing an item by index takes $O(1)$ worst-case time while grow and shrink operations take $O(r)$ amortized time. Using an exact analysis of a *growth game*, we show that for any data structure from a wide class of data structures that uses only $N + O(N^{1/r})$ space to store the array, the amortized cost of grow is $\Omega(r)$, even if only grow and access operations are allowed. The time for grow and shrink operations cannot be made worst-case, unless $r = 2$.

4 Open problems

4.1 Reducing weighted matching to unweighted matching

Aaron Bernstein (Rutgers University – New Brunswick, US)

License  Creative Commons BY 4.0 International license
© Aaron Bernstein

In dynamic unweighted matching, the goal is to maintain an (approximate) maximum cardinality matching in a graph that is changing over time. There is an extensive literature on this problem, with many different state-of-the-art results; these achieve different approximation-ratio/update-time tradeoffs, and also vary on secondary parameters (e.g. worst-case vs. amortized, adaptive vs. oblivious adversary, fully dynamic vs. decremental vs. incremental).

For almost all of these trade-offs, the state-of-the-art for maximum *weighted* matching lags far behind that for unweighted matching. One could try to adapt each unweighted algorithm separately to the weighted case, but it would be nice to have a single all-purpose tool.

There has been some partial progress in this direction. In 2017, Stubbs and Vassilevska Williams showed how to transform any algorithm for unweighted matching into one for weighted matching, but at the cost of a $(1/2 - \epsilon)$ approximation [2]. In 2021, Bernstein, Dudeja, and Langley reduced the approximation overhead to $2/3 - \epsilon$ in non-bipartite graphs and $1 - \epsilon$ in bipartite graphs [1]. That is, in bipartite graphs, this paper shows a black-box conversion from *any* algorithm for unweighted matching to one for weighted matching that is essentially as good: the approximation guarantee reduces by $1 - \epsilon$, while the update time increases by $\log(W)$. Moreover, this transformation preserves all secondary parameters (worst-case, deterministic, etc.); it also works in multiple other models (e.g. streaming, MPC).

The above algorithm gives us a blueprint for the kind of result we would like, but it only works in bipartite graphs. (In non-bipartite graphs the approximation overhead is $3/2$, which in matching is a big drawback.) Is it possible to get a similar transformation in non-bipartite graphs?

Open Problem

Is there a black-box transformation that converts any algorithm for dynamic unweighted matching in *non-bipartite* graphs into one for dynamic weighted matching, while only reducing the approximation guarantee by $(1 - \epsilon)$, and only increasing the update time by $\text{polylog}(nW)$?

References

- 1 Aaron Bernstein, Aditi Dudeja, and Zachary Langley. A framework for dynamic matching in weighted graphs. In *STOC*, pages 668–681. ACM, 2021.
- 2 Daniel Stubbs and Virginia Vassilevska Williams. Metatheorems for dynamic weighted matching. In *ITCS*, pages 58:1–58:14, 2017.

4.2 Communication Complexity of Max-Flow

Joakim Blikstad (KTH Royal Institute of Technology – Stockholm, SE)

License  Creative Commons BY 4.0 International license
© Joakim Blikstad

Suppose the edges of a graph G (with n vertices and m edges) are partitioned between two parties Alice and Bob. They wish to solve the (s, t) -max-flow problem on their graph with as few bits of communication as possible (in any number of interactive rounds).

Open Problem

It is an open problem to settle this communication complexity of max-flow. Specifically, are there near linear in n communication protocols, or can we show higher lower bounds? Another direction is looking at the round-communication tradeoff (relevant for streaming, distributed, etc.) of max-flow and related problems.

Some known “results”:

- Trivial algorithm $\tilde{O}(m)$: Alice sends all her edges to Bob.
- Interior point method $\tilde{O}(n\sqrt{n})$: One can simulate the $O(\sqrt{n})$ -round interior point method for max-flow in $O(n)$ communication each rounds. (*not verified, but should work*)
- $\Omega(n \log n)$ lower bound.

Evidence that a $\tilde{O}(n)$ communication protocol might / might not exist:

- The related problem of bipartite matching and its variants (max-cost b -matching, vertex capacitated s, t -flow, transshipment, negative weight SSSP) admit $\tilde{O}(n)$ communication protocol [1] (based on a straightforward cutting planes approach).
- The $\tilde{O}(n\sqrt{n})$ -communication interior point method shows that $O(m)$ is not the answer.
- Experience says that sequential $O(m^\alpha)$ graph algorithms usually translate into $O(n^\alpha)$ communication protocols. However, it is unclear if the sequential $O(m^{1+o(1)})$ time max-flow algorithm [2] would help in the communication setting (it uses $O(m)$ IPM rounds, and relies on complicated data structures).
- The answer can consist of up to $O(m)$ edges. So any $o(m)$ protocol cannot let the parties know the actual edges of the flow (but only a more “compact” representation). This is unlike all the bipartite-matching problems listed above that only use $O(n)$ edges in their optimal flows.

References

- 1 Joakim Blikstad, Jan van den Brand, Yuval Efron, Sagnik Mukhopadhyay, and Danupon Nanongkai. Nearly optimal communication and query complexity of bipartite matching. *CoRR*, abs/2208.02526, 2022. in FOCS'22.
- 2 Li Chen, Rasmus Kyng, Yang P. Liu, Richard Peng, Maximilian Probst Gutenberg, and Sushant Sachdeva. Maximum flow and minimum-cost flow in almost-linear time. *CoRR*, abs/2203.00671, 2022. in FOCS'22.

4.3 Dynamic Complexity of Low-Stretch Spanning Trees

Gramoz Goranci (ETH Zürich, CH)

License  Creative Commons BY 4.0 International license
 Gramoz Goranci

In this *low-stretch spanning tree* problem, we are given an undirected graph G with n vertices and m edges, and the goal is to compute a spanning tree T of G that minimizes $\text{av-str}_T(G) := (1/m) \sum_{e=(u,v) \in E(G)} \text{dist}_T(u,v)$, referred to as the *average stretch* of T . This problem and its variants lie at the core of algorithm design and have found applications in online and approximation algorithms, fast algorithms for computing maximum flows on graphs and solving Laplacian systems, and constructing competitive oblivious routing schemes, among others. It is known that every graph admits a spanning tree of average stretch $O(\log n \log \log n)$ which can be computed in $O(m \log n \log \log n)$ [1]. The stretch guarantee is tight up to a $O(\log \log n)$ factor since any spanning tree of a n -vertex grid graph requires $\Omega(\log n)$ average stretch [2].

In the fully dynamic setting, the graph G undergoes an intermixed sequence of edge insertions and deletions, and the goal is to maintain a spanning tree T of G with small average stretch. The first work in this setting [4] gave a dynamic algorithm that supports edge updates in $n^{1/2+o(1)}$ amortized update and ensures that the maintained spanning tree has average stretch $n^{o(1)}$. The update time was subsequently improved to $n^{o(1)}$ while keeping the stretch guarantee the same [3]. Both works make use of randomization and assume an oblivious adversary.

Open Problems

- Does there exist a fully dynamic algorithm for maintaining a spanning tree with $O(\text{poly}(\log n))$ average stretch in *sub-linear* update time? The question is interesting even in the decremental/deletions-only setting.
- Are there *deterministic* or randomized algorithms that work against *adaptive adversaries* for the dynamic low-stretch spanning tree problem?

References

- 1 Ittai Abraham and Ofer Neiman. Using petal-decompositions to build a low stretch spanning tree. *SIAM J. Comput.*, 48(2):227–248, 2019.
- 2 Noga Alon, Richard M. Karp, David Peleg, and Douglas B. West. A graph-theoretic game and its application to the k-server problem. *SIAM J. Comput.*, 24(1):78–100, 1995.
- 3 Shiri Chechik and Tianyi Zhang. Dynamic low-stretch spanning trees in subpolynomial time. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 463–475. SIAM, 2020.

- 4 Sebastian Forster and Gramoz Goranci. Dynamic low-stretch trees via dynamic low-diameter decompositions. In Moses Charikar and Edith Cohen, editors, *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*, pages 377–388. ACM, 2019.

4.4 Polylog query time for a dynamic all-pairs problem in plane directed graphs

Adam Karczmarz (University of Warsaw – Warsaw, PL & and IDEAS NCBR – Warsaw, PL)

License  Creative Commons BY 4.0 International license
© Adam Karczmarz

Plane directed graphs allow non-trivial dynamic reachability and distance oracles supporting arbitrary point-to-point queries. For example, one can achieve $\tilde{O}(\sqrt{n})$ update/query time bound for fully dynamic reachability [2], or incremental distances [1], and $\tilde{O}(n^{2/3})$ update/query time for fully dynamic distances [3]. For decremental reachability, one can get $\text{polylog}(n)$ amortized update time and $\tilde{O}(\sqrt{n})$ query time [4]. However, to the best my knowledge, no tradeoff with $\text{polylog}(n)$ query time and $\tilde{O}(n^{0.99})$ amortized update time is known for any kind of dynamic all-pairs oracle problem on plane digraphs. Probably the easiest specific problem addressing this should be the following.

Open Problem

Suppose a plane digraph G is given. Initially, all edges are switched off, and G undergoes edge switch-ons. Design a data structure supporting $\text{polylog}(n)$ -time arbitrary-pair reachability queries (in the switched on subgraph) and edge switch-ons within $\tilde{O}(n^{1.99})$ total update time.

References

- 1 Debarati Das, Maximilian Probst Gutenberg, and Christian Wulff-Nilsen. A near-optimal offline algorithm for dynamic all-pairs shortest paths in planar digraphs. In *Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms, SODA 2022, Virtual Conference / Alexandria, VA, USA, January 9 – 12, 2022*, pages 3482–3495. SIAM, 2022.
- 2 Krzysztof Diks and Piotr Sankowski. Dynamic plane transitive closure. In *Algorithms – ESA 2007, 15th Annual European Symposium, Eilat, Israel, October 8-10, 2007, Proceedings*, volume 4698 of *Lecture Notes in Computer Science*, pages 594–604. Springer, 2007.
- 3 Jittat Fakcharoenphol and Satish Rao. Planar graphs, negative weight edges, shortest paths, and near linear time. *J. Comput. Syst. Sci.*, 72(5):868–889, 2006.
- 4 Adam Karczmarz. Decremental transitive closure and shortest paths for planar digraphs and beyond. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 73–92. SIAM, 2018.

4.5 Breaking CountMin Sketches Inside a Greedy Outer Loop

Richard Peng (University of Waterloo, CA)

License © Creative Commons BY 4.0 International license
© Richard Peng

Consider the following way of estimating cardinalities of subsets of $[n] = \{1 \dots n\}$:

1. Pick a random permutation π of $1, 2, \dots, n$.
2. For a set $S \subseteq [n]$ with size at least $1000 \log n$, store the smallest $100 \log n$ values of

$$\pi(S) = \{\pi(i) : i \in S\},$$

and take their max.

Call this value $Sketch_\pi(S)$.

It can be shown using a reasonably ‘standard’ use of Chernoff bound that if $|S_1| < 2|S_2|$, then with probability at least $1 - n^{-3}$ (over the choices of π), $Sketch_\pi(S_1) < Sketch_\pi(S_2)$. Also, as the sketches have size $O(\log n)$, such a schema gives a low storage method for approximating tracking sizes of sets under mergers. The fun, then happens when one starts to use the output of the data structures to dictate the next merge. That is, consider starting with n sets $S_1 \dots S_n$ (of size at least $1000 \log n$, which is easy to enforce by having ‘dummy’ elements that are in all sets), and after generating an initial random permutation π , repeatedly perform the following simplification of the min-degree heuristic:

1. For $t = 1 \dots n - 2$
 - a. Let i and j be the two remaining sets with the minimum / second minimum sketch values computed w.r.t. π .
 - b. Replace S_i, S_j in the collection of sets by their union, $S_i \cup S_j$.

Open Problem

Exhibit an initial state such that with probability at least 0.1 over the choices of π , at some iteration of the algorithm, one of S_i and S_j has size more than twice the minimum / second minimum respectively. Alternatively, prove this cannot happen, that is, for any starting configurations of S_1, S_2, \dots, S_n , things are happy with probability > 0.1 (over choices of π).

A stronger form of the latter version is showing that in absences of deletions, so just queries and merges, things work.

References:

- Paper that showed approximate min-degree orderings can be solved in almost-linear time by re-introducing randomness to decorrelate intermediate states: <https://arxiv.org/abs/1804.04239>.
- The analysis of randomized Gaussian elimination (which does combine such randomness against adversarial users) by Kyng and Sachdeva: <https://arxiv.org/abs/1605.02353>, and a follow up that defined a ‘resparsification game’: <https://arxiv.org/abs/1611.06940>.

4.6 Dynamic Maximal Matching

Shay Solomon (Tel Aviv University, IL)

License  Creative Commons BY 4.0 International license
© Shay Solomon

In the dynamic *maximal matching* (MM) problem, the goal is to maintain a maximal matching in a dynamic n -vertex graph that is subject to edge updates. There is a naive deterministic algorithm with a worst-case update time of $O(n)$, and this is the state-of-the-art update time of deterministic algorithms in general graphs, and also of randomized algorithms against an adaptive adversary, even allowing amortization.

It should be noted that in some graph classes, better results are known. In particular, for sparse graphs, there is a deterministic algorithm [2] with a worst-case update time of $O(\sqrt{m})$, where m is the dynamic number of edges in the graph.

This problem provides a prime example of an exponential gap between algorithms that cope against oblivious versus adaptive adversaries. Indeed, allowing randomization against an oblivious adversary, one can achieve a constant amortized update time [3] and a poly-log worst-case update time [1]. We also note that, while the $O(n)$ (or $O(\sqrt{m})$) deterministic bound has resisted any improvement, no lower bound whatsoever is known.

Open Problem

Is there a deterministic algorithm, or a randomized one against an adaptive adversary, for maintaining MM in $o(n)$ update time in general graphs? Can we push the update time towards a poly-log or even further? Is there any $\omega(1)$ lower bound for such algorithms?

References

- 1 Aaron Bernstein, Sebastian Forster, and Monika Henzinger. A deamortization approach for dynamic spanner and dynamic maximal matching. In *30th SODA*, pages 1899–1918. SIAM, 2019.
- 2 Ofer Neiman and Shay Solomon. Simple deterministic algorithms for fully dynamic maximal matching. In *STOC*, pages 745–754, 2013.
- 3 Shay Solomon. Fully dynamic maximal matching in constant update time. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 325–334, 2016.

4.7 Dynamic Derandomization

David Wajc (Stanford University, US)

License  Creative Commons BY 4.0 International license
© David Wajc

A common theme in derandomization is the construction of small near-independent probability spaces. The following definition, due to Naor and Naor [1], captures (some) such notions, and generalizes the perhaps more familiar notion of k -wise independence.

Definition. Let \mathcal{U} be the uniform distribution on n binary variables. A distribution \mathcal{D} on n binary variables is (δ, k) -dependent if for any event A that is determined by k or fewer

random variables,

$$\Pr_{\bar{Y} \sim \mathcal{U}} [A] - \delta \leq \Pr_{\bar{Y} \sim \mathcal{D}} [A] \leq \Pr_{\bar{Y} \sim \mathcal{U}} [A] + \delta.$$

Equivalently, \mathcal{D} satisfies that for any subset of k or fewer indices, $I \subseteq [m]$, $|I| \leq k$,

$$\sum_{\vec{v} \in \{0,1\}^{|I|}} \left| \Pr \left[\bigwedge_{i \in I} (Y_i = v_i) \right] - 2^{-|I|} \right| \leq \delta.$$

Similarly, up to constants, another way of stating the above is that the total variation distance between the distribution of any subset of k variables from the uniform distribution is $O(\delta)$. For example, k -wise independent distributions, under which any k variables take on all 2^k realizations with probability 2^{-k} , are precisely $(0, k)$ -dependent distributions. The interest in these distributions is due to the following lemma of [1].

Lemma. *For any $\delta > 0$, a (δ, k) -dependent distribution \mathcal{D} on n binary variables can be constructed using $\log \log n + O(k + \log(\frac{1}{\delta}))$ random bits.¹ Moreover, after polytime preprocessing, each random variable can be sampled from \mathcal{D} in $O(k \cdot \log n)$ time.*

The utility of the above for derandomization should be apparent: given a randomized algorithm whose analysis hinges on events determined by few random variables, and allows for small error compared to a fully independent distribution, we can simply try out all random seeds!

In more detail, given an algorithm \mathcal{A} using n random bits, but whose analysis carries through unchanged with (δ, k) -dependent variables, construct a deterministic algorithm \mathcal{A}' that runs a copy of \mathcal{A} for all $2^{\log \log n + O(k + \log(1/\delta))} = \log n \cdot 2^{O(k) + \log(1/\delta)}$ many random seeds for the appropriate distribution. So, for example, for constant k and δ , the obtained deterministic algorithm \mathcal{A}' is only slower than \mathcal{A} by some polynomial additive term (to build \mathcal{D}) and a polylogarithmic multiplicative factor (due to trying all seeds for \mathcal{D}).

The above approach has found applications in derandomization numerous areas since its introduction. A non-exhaustive list of application areas include parallel computing [8], streaming [7], local computation [6], the Color Coding technique [5], matrix multiplication witnesses [4], and recently in (partially derandomizing) online algorithms [3].

Conspicuously missing from the above list (and its extensions) is the area of dynamic algorithms, where derandomizing via this approach would incur a logarithmic overhead (for sufficiently small k and δ) by running one copy of the random algorithm for each random seed. This suggests the following open question.

Open Problem

Find an application of (δ, k) -dependent distributions in derandomizing dynamic algorithms, possibly with only a polylogarithmic overhead.

References

- 1 Joseph (Seffi) Naor and Moni Naor. Small-bias probability spaces: Efficient constructions and applications. *SICOMP*, 22(4):838–856, 1993.
- 2 A Joffe. On a set of almost deterministic k -independent random variables. *the Annals of Probability*, 2(1):161–162, 1974.

¹ For k -wise independence ($\delta = 0$), the necessary number of random bits blows up to $\lceil k/2 \rceil \log n$ [2].

- 3 Niv Buchbinder, Joseph Seffi Naor, and David Wajc. Lossless online rounding for online bipartite matching (despite its impossibility). In *SODA, 2023*. To appear.
- 4 Noga Alon and Moni Naor. Derandomization, witnesses for boolean matrix multiplication and construction of perfect hash functions. *Algorithmica*, 16(4):434–449, 1996.
- 5 Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *J.ACM*, 42(4):844–856, 1995.
- 6 Omer Reingold and Shai Vardi. New techniques and tighter bounds for local computation algorithms. *Journal of Computer and System Sciences*, 82(7):1180–1200, 2016.
- 7 Joan Feigenbaum, Yuval Ishai, Tal Malkin, Kobbi Nissim, Martin J Strauss, and Rebecca N Wright. Secure multiparty computation of approximations. In *International Colloquium on Automata, Languages, and Programming*, pages 927–938, 2001.
- 8 Noga Alon. A parallel algorithmic version of the local lemma. *Random Structures & Algorithms*, 2(4):367–378, 1991.

Participants

- Amir Abboud
Weizmann Institute – .
Rehovot, IL
- Aaron Bernstein
Rutgers University – New
Brunswick, US
- Sayan Bhattacharya
University of Warwick –
Coventry, GB
- Joakim Blikstad
KTH Royal Institute of
Technology – Stockholm, SE
- Karl Bringmann
Universität des Saarlandes –
Saarbrücken, DE
- Keren Censor-Hillel
Technion – Haifa, IL
- Shiri Chechik
Tel Aviv University, IL
- Keerti Choudhary
Indian Institute of Technology –
New Dehli, IN
- Aleksander Christiansen
Technical University of Denmark
– Lyngby, DK
- Jeremy Fineman
Georgetown University –
Washington, DC, US
- Nick Fischer
MPI für Informatik –
Saarbrücken, DE
- Sebastian Forster
Universität Salzburg, AT
- Pawel Gawrychowski
University of Wrocław, PL
- Gramoz Goranci
ETH Zürich, CH
- Fabrizio Grandoni
SUPSI – Lugano, CH
- Kathrin Hanauer
Universität Wien, AT
- Adam Karczmarz
University of Warsaw, PL & and
IDEAS NCBR – Warsaw, PL
- Peter Kiss
University of Warwick –
Coventry, GB
- Tsvi Kopelowitz
Bar-Ilan University –
Ramat Gan, IL
- William Kuszmaul
MIT – Cambridge, US
- Jakub Lacki
Google – New York, US
- Nicole Megow
Universität Bremen, DE
- Matthias Mnich
TU Hamburg, DE
- Shay Mozes
Reichman University –
Herzliya, IL
- Danupon Nanongkai
MPI für Informatik –
Saarbrücken, DE
- Yasamin Nazari
Universität Salzburg, AT
- Nikos Parotsidis
Google Research – Zürich, CH
- Richard Peng
University of Waterloo, CA
- Maximilian Probst Gutenberg
ETH Zürich, CH
- Eva Rotenberg
Technical University of Denmark
– Lyngby, DK
- Piotr Sankowski
IDEAS NCBR – Warsaw, PL &
University of Warsaw, PL
- Thatchaphol Saranurak
University of Michigan –
Ann Arbor, US
- Christian Schulz
Universität Heidelberg, DE
- Chris Schwiegelshohn
Aarhus University, DK
- Shay Solomon
Tel Aviv University, IL
- Clifford Stein
Columbia University, US
- David Tench
Rutgers University –
Piscataway, US
- Jan van den Brand
Georgia Institute of Technology –
Atlanta, US
- Virginia Vassilevska Williams
MIT – Cambridge, US
- David Wajc
Stanford University, US
- Oren Weimann
University of Haifa, IL
- Nicole Wein
Rutgers University –
Piscataway, US
- Uri Zwick
Tel Aviv University, IL



Set Visualization and Uncertainty

Susanne Bleisch^{*1}, Steven Chaplick^{*2}, Jan-Henrik Haunert^{*3},
Eva Mayr^{*4}, Marc van Kreveld^{*5}, and Annika Bonerath^{†6}

- 1 FH Nordwestschweiz – Muttenz, CH. susanne.bleisch@fhnw.ch
- 2 Maastricht University, NL. s.chaplick@maastrichtuniversity.nl
- 3 Universität Bonn, DE. haunert@igg.uni-bonn.de
- 4 Donau-Universität Krems, AT. eva.mayr@donau-uni.ac.at
- 5 Utrecht University, NL. m.j.vankreveld@uu.nl
- 6 Universität Bonn, DE. bonerath@igg.uni-bonn.de

Abstract

The Dagstuhl Seminar on Set Visualization and Uncertainty brought together a group of researchers from diverse disciplines, all of which are interested in various aspects of this type of visualization: the cognitive aspects, the modelling aspects, the algorithmic aspects, and the information visualization aspects. An important but difficult to handle problem is how one should visualize information with underlying uncertainty. The seminar focused on uncertainty in set systems. This report includes short abstracts of the talks given during the seminar as well as more extensive working group reports on the research done during the seminar.

Seminar November 13–18, 2022 – <http://www.dagstuhl.de/22462>

2012 ACM Subject Classification Human-centered computing → Visualization; Theory of computation → Design and analysis of algorithms

Keywords and phrases cartography, graph drawing, information visualization, set visualization, uncertainty

Digital Object Identifier 10.4230/DagRep.12.11.66

1 Executive Summary


Susanne Bleisch (FH Nordwestschweiz – Muttenz, CH)

Steven Chaplick (Maastricht University, NL)

Jan-Henrik Haunert (Universität Bonn, DE)

Eva Mayr (Donau-Universität Krems, AT)

Marc van Kreveld (Utrecht University, NL)

License  Creative Commons BY 4.0 International license

© Susanne Bleisch, Steven Chaplick, Jan-Henrik Haunert, Eva Mayr, and Marc van Kreveld

Research Area

The topic of *Set Visualization and Uncertainty* is inherently interdisciplinary, combining aspects of several diverse fields. As such, the overview of the research area is split into the key fields associated with it; namely, information visualization, set systems, graph drawing, uncertainty (as applied to data sets), and cartography.

Information visualization (InfoVis) can help humans gain insight from large volumes of data by providing good graphical overviews as well as appropriate interfaces for accessing details (see, e.g., [1]). It has thus become of high relevance for industry and many scientific

* Editor / Organizer

† Editorial Assistant / Collector



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

Set Visualization and Uncertainty, *Dagstuhl Reports*, Vol. 12, Issue 11, pp. 66–95

Editors: Susanne Bleisch, Steven Chaplick, Jan-Henrik Haunert, Eva Mayr, Marc van Kreveld, and Annika Bonerath



DAGSTUHL REPORTS Dagstuhl Reports

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

disciplines. Since the generation of effective visualizations requires knowledge of human cognition, algorithms, data characteristics, visual variables, and tasks, the InfoVis community embraces members of various disciplines, including computer scientists of different areas, cognitive scientists, psychologists, and cartographers.

Sets are mathematically defined as unordered collections of distinct objects. They play an important role in InfoVis since reasoning based on aggregated information (i.e., sets instead of individual objects) can greatly reduce the complexity of data analysis tasks. Most often, the sets are defined by categories of objects; e.g., people can be grouped by country of residence, education, or gender to study influences on income. Often, the aim is to visualize statistics (e.g., number of elements, average income) for each set and, since an element can be member of multiple sets, the relationships between them (e.g., intersection and containment). Set visualization is traditionally done with Venn or Euler diagrams, yet a plethora of alternative visualization types for sets has been developed. A recent focus of research has been on developing scalable solutions (e.g., to create effective visualizations for very large set systems) and dealing with dynamics (e.g., changes of the elements' set memberships over time). In this seminar we dealt with a different issue, already relevant for static and small set systems: uncertainty. Although the importance of uncertainty visualization has been stressed by several researchers, only few studies exist that deal with it specifically in the context of sets and systems of sets [3].

Uncertainty is inherent to almost any information collected through observations by humans or sensors. Since the assignment of elements to categories corresponding to sets follows observations, the set memberships are uncertain, too. Moreover, subsuming multiple elements with their individual properties under one category results in a loss of information. Although this information reduction may be intended to reduce the graphical complexity, visualizing the within-set as well as the between-set variability may improve the interpretation of the data. Uncertainty is usually evaluated with statistical methods or concepts of probability. Uncertainty can relate to the existence of an element, the existence of a set, the presence of an element in a set, set containment in hierarchies, location of an object in geo-located data, etc. Moreover, uncertainty can be given as a binary property or as a probability. Fuzzy set theory extends the idea of sets by allowing partial set membership, indicated by a value between 0 and 1. This model has been proposed for concepts that lack crisp boundaries (e.g., “young” and “old” as categories of people). In InfoVis, an important question is whether and how the uncertainty of the information displayed should be visually encoded (e.g., with glyphs or graphical variables), and how users process this visualization of uncertainty [2]. Moreover, although the uncertainty may not be depicted, it may be considered when generating a visualization (e.g., by filtering information based on its certainty). Conversely, a standard visualization like a heat map suggests uncertainty which may not exist in the data at all.

Graph drawing is a branch of computer science focusing on the computation of geometric layouts of graphs, involving both formal and experimental methods. Since graphs are useful mathematical models for networks, graph drawing is of high relevance for network visualization. Graph drawing can be applied to non-geometric networks (e.g., social networks consisting of friendship relationships) as well as to geometric networks (e.g., networks of metro lines) if the aim is to generate more abstract (e.g., schematic) representations. Since a system of sets can be considered as a hypergraph in which each node corresponds to one entity and each hyperedge corresponds to one set, set visualization is fundamental in the graph drawing community. However, aspects of uncertainty remain mostly unexplored [4].

Geographic information is a combination of geometric, temporal, and attribute information, each of which can be uncertain in different ways and can be visualized in different ways commonly through maps. *Cartography* and its sister discipline *Geographic Information Science* have a long history in dealing with uncertainty in the context of analyzing and visualizing spatial information. For example, international standards formalizing elements of spatial data quality have been established (e.g., ISO 19157:2013 defining thematic accuracy, temporal quality, positional accuracy) and graphical variables encoding the uncertainty of information in maps have been proposed, including color saturation and symbol focus [5].

Seminar Goals

This seminar aimed to advance research into methods and techniques for set visualizations and uncertainty by fostering interdisciplinary and cross-domain collaboration (cf. section Research Area). Sets are mathematically defined as collections of distinct objects. They play an important role in Information Visualization since reasoning based on aggregated information can reduce the complexity of the analysis tasks. Uncertainty is inherent to almost any information collected through observations by humans or sensors and, thus, also set elements or their set membership. Uncertainty generally adds to the complexity of data analysis and data presentation. In the seminar we looked specifically into approaches for dealing with uncertain information when visualizing sets. Information Visualization has some established techniques regarding uncertainty. However, the topic is – except for some specific cases (cf. Fig. 1) – mostly unexplored in the context of set visualizations. Some uncertainty visualization techniques may directly apply to set visualizations. In this seminar we brought together researchers from the areas of information visualization, visual analytics, graph drawing, geoinformation science, uncertainty research, and cognitive science. These interdisciplinary participants formed working groups to consider selected problems of considering and visualising uncertainty associated with sets so that the visualizations are informative and reliable, in the sense that humans can use them for visual analysis tasks and that the uncertain information is recognizable.

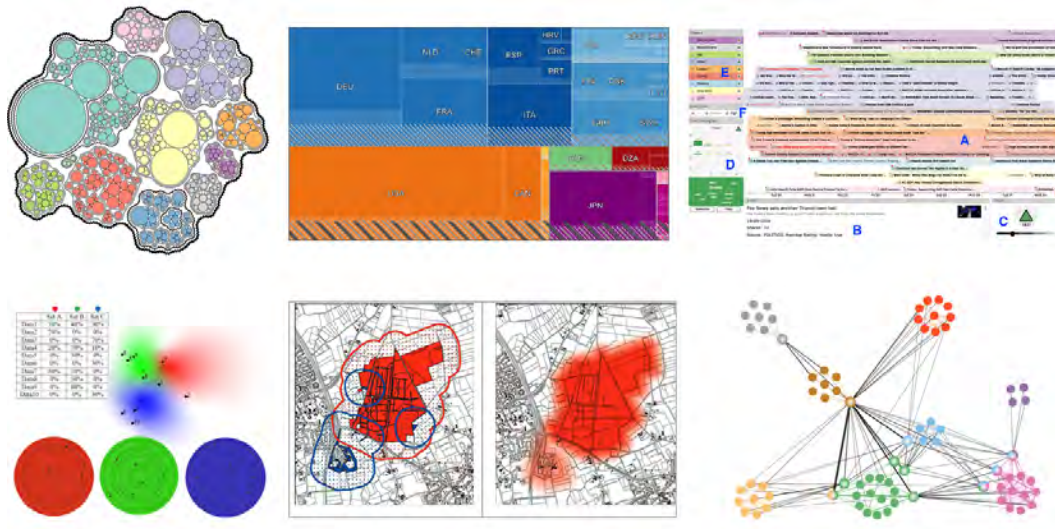
Seminar Format

The interdisciplinary topic of the seminar, as well as the different scientific backgrounds of the participants, asked for an introduction to the main topics as well as to selected perspectives through invited talks on the first day. The structure of two talks in the morning and two in the afternoon of the first day left enough room for first discussions. The day ended with participants' pitches of open problems and the participants indicating their interest in the pitched problems.

Invited talks of the first day:

- Daniel Archambault: Drawing Euler Diagrams with Closed Curves
- Wouter Meulemans: Algorithmic Perspectives on Uncertainty and Set Visualization
- Bei Wang Phillips: Visualizing Hypergraphs: With Connections to Uncertainty Visualization
- Martin Kryzwinski: Genomes: sets of sets of sets

The second day of the seminar was started with the formation of four groups interested in four different open problems. Each group worked on their specific open problem for



■ **Figure 1** Examples of uncertainty visualizations for sets from [6, 7, 8, 9, 10, 11] (from top left to bottom right).

the remainder of the seminar. Participants were invited to give mini-talks related to the seminar topic. Time was reserved for those contributed talks every morning. Additionally, the working groups reported on their progress on Wednesday and Friday.

Contributed mini-talks throughout the week (given are the names of the presenters, see Overview of Talks for full list of contributors):

- Annika Bonerath & Markus Wallinger: MosaicSets
- Sara Irina Fabrikant: How to visualize uncertainty
- Silvia Miksch: Visual Encodings of Temporal Uncertainty: A Comparative User Study
- Nathan van Beusekom: Simultaneous Matrix Orderings for Graph Collections
- Marc van Kreveld: On Full Diversity in Metric Spaces
- Alexander Wolff: StoryLines

Outcomes and Future Plans

The participants were highly satisfied with the quality of the seminar. Diverse interdisciplinary discussions took place and all groups worked well together. The final progress reports of the working groups indicate that the collaborations will be ongoing and some papers will be published (cf. section Working Groups).

At the final day plenary meeting, plans for a follow-up seminar were discussed. A group of interested participants is currently discussing the focus and title of such a seminar.

References

- 1 Munzner, T. (2014). *Visualization Analysis & Design*. Boca Raton: CRC Press.
- 2 Padilla, L. and Kay, M. and Hullman, J. (2021). Uncertainty Visualization in *Wiley StatsRef: Statistics Reference Online*. Eds. N. Balakrishnan and T. Colton and B. Everitt and W. Piegorsch and F. Ruggeri and J.L. Teugels, Wiley. doi:10.1002/9781118445112.
- 3 Alsallakh, Bilal and Micallef, Luana and Aigner, Wolfgang and Hauser, Helwig and Miksch, Silvia and Rodgers, Peter (2016). The State-of-the-Art of Set Visualization. *Computer Graphics Forum*, 35(1):234–260. doi:10.1111/cgf.12722.

- 4 Steven Chaplick and Takayuki Itoh and Giuseppe Liotta and Kwan-Liu Ma and Ignaz Rutter (2020). *Trends and Perspectives in Graph Drawing and Network Visualization*, NII Shonan Meeting Report 171. <https://shonan.nii.ac.jp/docs/342e2e86373d4783ae00063ddd67e8b35b9d75d5.pdf>.
- 5 MacEachren, Alan M. and Robinson, Anthony and Hopper, Susan and Gardner, Steven and Murray, Robert and Gahegan, Mark and Hetzler, Elisabeth (2005). Visualizing Geospatial Information Uncertainty: What We Know and What We Need to Know. *Cartography and Geographic Information Science*, 32(3):139–160. doi:10.1559/1523040054738936.
- 6 Görtler, Jochen and Schulz, Christoph and Weiskopf, Daniel and Deussen, Oliver (2018). Bubble Treemaps for Uncertainty Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):719–728. doi:10.1109/TVCG.2017.2743959.
- 7 Sondag, Max and Meulemans, Wouter and Schulz, Christoph and Verbeek, Kevin and Weiskopf, Daniel and Speckmann, Bettina (2020). Uncertainty Treemaps. *2020 IEEE Pacific Visualization Symposium (PacificVis)*, 111–120. IEEE.
- 8 Xu, Kai and Salisu, Saminu and Nguyen, Phong H. and Walker, Rick and Wong, B. L. William and Wagstaff, Adrian and Phillips, Graham and Biggs, Mike and Potel, Mike (2020). TimeSets: Temporal Sensemaking in Intelligence Analysis. *IEEE Computer Graphics and Applications*, 40(3):83–93. doi:10.1109/MCG.2020.2981855.
- 9 Zhu, Lifeng and Xia, Weiwei and Liu, Jia and Song, Aiguo (2018). Visualizing fuzzy sets using opacity-varying freeform diagrams. *Information Visualization*, 17(2):146–160. doi:10.1177/1473871617698517.
- 10 Vullings, L. A. E. and Blok, C. A. and Wessels, C. G. A. M. and Bulens, J. D. (2013). Dealing with the uncertainty of having incomplete sources of geo-information in spatial planning. In *Applied spatial analysis and policy*, 6(1):25–45. Springer.
- 11 Vehlow, Corinna and Reinhardt, Thomas and Weiskopf, Daniel (2013). Visualizing Fuzzy Overlapping Communities in Networks. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2486–2495. doi:10.1109/TVCG.2013.232.

2 Table of Contents

Executive Summary

Susanne Bleisch, Steven Chaplick, Jan-Henrik Haurert, Eva Mayr, and Marc van Kreveld 66

Overview of Talks

(Invited) Drawing Euler Diagrams with Closed Curves
Daniel Archambault 72

(Invited) Algorithmic Perspectives on Uncertainty and Set Visualization
Wouter Meulemans 72

(Invited) Visualizing Hypergraphs: With Connections to Uncertainty Visualization
Bei Wang Phillips 73

(Invited) Genomes: sets of sets of sets
Martin Krzywinski 73

(Contributed) MosaicSets
Annika Bonerath, Sven Gedicke, Jan-Henrik Haurert, Martin Nöllenburg, Peter Rottmann, and Markus Wallinger 74

(Contributed) How to visualize uncertainty
Sara Irina Fabrikant 75

(Contributed) Visual Encodings of Temporal Uncertainty: A Comparative User Study
Silvia Miksch 76

(Contributed) Simultaneous Matrix Orderings for Graph Collections
Nathan Van Beusekom and Wouter Meulemans 76

(Contributed) On Full Diversity in Metric Spaces
Marc van Kreveld 77

(Contributed) StoryLines
Alexander Wolff 77

Working groups

StorySets
Annika Bonerath, Stephen G. Kobourov, Wouter Meulemans, Martin Nöllenburg, Markus Wallinger, and Alexander Wolff 78

Simplification of Set Systems to Draw Nice Euler Diagrams
Peter Rodgers, Daniel Archambault, Jan-Henrik Haurert, and Bei Wang Phillips . 83

Types of Uncertainty in Set Visualization
Christian Tominski, Michael Behrisch, Susanne Bleisch, Sara Irina Fabrikant, Eva Mayr, Silvia Miksch, and Helen C. Purchase 85



Set Size Visualization with Dependent and Independent Uncertainties
Nathan Van Beusekom, Steven Chaplick, Amy Griffin, Martin Krzywinski, Marc van Kreveld, and Hsiang-Yun Wu 90

Participants 95

3 Overview of Talks

3.1 (Invited) Drawing Euler Diagrams with Closed Curves

Daniel Archambault (Swansea University, GB)

License  Creative Commons BY 4.0 International license
 Daniel Archambault

Joint work of Paolo Simonetto, Daniel Archambault, Carlos Scheidegger, David Auber

Main reference Paolo Simonetto, Daniel Archambault, Carlos Scheidegger: “A Simple Approach for Boundary Improvement of Euler Diagrams”, *IEEE Trans. Vis. Comput. Graph.*, Vol. 22(1), pp. 678–687, 2016.
URL <https://doi.org/10.1109/TVCG.2015.2467992>

Main reference Paolo Simonetto, David Auber, Daniel Archambault: “Fully Automatic Visualisation of Overlapping Sets”, *Comput. Graph. Forum*, Vol. 28(3), pp. 967–974, 2009.

URL <https://doi.org/10.1111/j.1467-8659.2009.01452.x>

Main reference Paolo Simonetto, Daniel Archambault, David Auber, Romain Bourqui: “ImPrEd: An Improved Force-Directed Algorithm that Prevents Nodes from Crossing Edges”, *Comput. Graph. Forum*, Vol. 30(3), pp. 1071–1080, 2011.

URL <https://doi.org/10.1111/j.1467-8659.2011.01956.x>

One of the typical methods for visualising sets is through Euler diagrams represented as closed curves. In this talk, I recap some work on force-directed drawings of Euler diagrams and the scalability of such methods. In particular, I speak of force-directed methods for drawing Euler diagrams and methods for refining them given a drawing. I conclude with some open problems that involve representing uncertainty in this representation.

3.2 (Invited) Algorithmic Perspectives on Uncertainty and Set Visualization

Wouter Meulemans (TU Eindhoven, NL)

License  Creative Commons BY 4.0 International license
 Wouter Meulemans

Treemaps are a common way to visualize hierarchical numeric data (file systems, census data, economic data). However, in many cases, the numeric values have associated uncertainty: arising for example from the data-collection process or from data transformations like aggregation over time. I will discuss a method [1] for creating treemaps that show both the data itself and the uncertainty, while maintaining the partitioning nature of treemaps. Then, I continue briefly to consider an alternative perspective: to artificially induce uncertainty to improve visual structure. Specifically, we will look at using spatial deformation to schematize set visualization [2] and relate this to research in algorithmic imprecision.

References

- 1 M. Sondag, W. Meulemans, C. Schulz, K. Verbeek, D. Weiskopf, and B. Speckmann. Uncertainty treemaps. In *Proceedings of the 2020 IEEE Pacific Visualization Symposium*, pages 111–120, 2020.
- 2 M. A. Bekos, D. J. C. Dekker, F. Frank, W. Meulemans, P. Rodgers, A. Schulz, and S. Wessel. Computing Schematic Layouts for Spatial Hypergraphs on Concentric Circles and Grids. *Computer Graphics Forum*, 41(6):316–335, 2022.

3.3 (Invited) Visualizing Hypergraphs: With Connections to Uncertainty Visualization

Bei Wang Phillips (University of Utah – Salt Lake City, US)

License © Creative Commons BY 4.0 International license

© Bei Wang Phillips

Joint work of Bei Wang Phillips, Youjia Zhou, Archit Rathore, Emilie Purvine, Samir Chowdhury, Tom Needham, Ethan Semrad

Main reference Youjia Zhou, Archit Rathore, Emilie Purvine, Bei Wang: “Topological Simplifications of Hypergraphs”, *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 2022.

URL <https://doi.org/10.1109/tvcg.2022.3153895>

Main reference Samir Chowdhury, Tom Needham, Ethan Semrad, Bei Wang, Youjia Zhou: “Hypergraph Co-Optimal Transport: Metric and Categorical Properties”, *CoRR*, Vol. abs/2112.03904, 2021.

URL <https://arxiv.org/abs/2112.03904>

In this talk, I first give a brief overview of hypergraph visualization, which is closely related to set visualization. Following the recent survey of Fischer et al. [1], hypergraph visualization techniques could be classified as node-link-based, matrix-based, and timeline-based approaches. During the overview, I ask the following questions with respect to uncertainty visualization: Where are the uncertainties? And how to encode uncertainties? I then discuss the current and future directions on hypergraph visualization. Specifically, from existing perspectives:

- Scalability;
- Aggregating and subsetting;
- Providing support for dynamic hypergraphs with a large number of time steps;
- Benchmark dataset for hypergraph visualizations;
- Performance metrics.

And from my own perspectives:

- Hypergraph simplification using topological approaches;
- Transforming hypergraphs to graphs while preserving metric structures;
- Hypergraph matching using measure theory and optimal transport;
- Uncertainty visualization for hypergraph ensembles.

References

- 1 M. T. Fischer, A. Frings, D. A. Keim, and D. Seebacher. Towards a survey on static and dynamic hypergraph visualizations. In *2021 IEEE visualization conference (VIS)*, pages 81–85, 2021.

3.4 (Invited) Genomes: sets of sets of sets

Martin Krzywinski (BC Cancer Research Centre – Vancouver, CA)

License © Creative Commons BY 4.0 International license

© Martin Krzywinski

The genomes in our cells naturally vary between individuals. These changes are mostly differences (mutations) at single base pair positions — two random individuals vary at about 3,000,000 locations (1 in 1,000). An individual may have an inherited mutation that predisposes them to disease (e.g. cancer) or have accumulated unrepaired DNA damage from environmental exposure (e.g. sunlight) that equally raises their risk. A mutation that triggers the onset of a disease is called a “driver mutation”. Such mutations typically dysregulate the repair systems of the genome and lead to an accumulation of errors – the genome becomes fragile and the cell may begin to divide without limitation imposed by checkpoints. As this

cell divides, mutations accumulate and now the tumor becomes a collection of groups of cells (clones), each with a slightly different genome. This process leads to a “family tree”, in which nodes are groups of cells.

Thus, cancers can be thought of as sets (individuals with a tumor) of sets (tumors composed of cell groups) of sets (cell groups composed of cells). Visualizing this complexity is challenging for the researcher (tools are only now appearing) and reader (the researcher typically lacks design and visualization experience). Good practices in the use of color, symbol and encoding, which are well known in the visualisation community, have not penetrated the cancer research community — which has only a vague (or no) awareness of best practices. The biologists are not versed in breaking down and addressing the challenges in creating complex visualisations.

I present case studies from the field of genomics and cancer research that illustrate common errors in visualisations in that field, show how I address them (on an individual basis) and identify areas in which visualisation and set community can contribute.

3.5 (Contributed) MosaicSets

Annika Bonerath (Universität Bonn, DE), Sven Gedicke, Jan-Henrik Haurert (Universität Bonn, DE), Martin Nöllenburg (TU Wien, AT), Peter Rottmann, and Markus Wallinger (TU Wien, AT)

License © Creative Commons BY 4.0 International license

© Annika Bonerath, Sven Gedicke, Jan-Henrik Haurert, Martin Nöllenburg, Peter Rottmann, and Markus Wallinger

Main reference Peter Rottmann, Markus Wallinger, Annika Bonerath, Sven Gedicke, Martin Nöllenburg, Jan-Henrik Haurert: “MosaicSets: Embedding Set Systems into Grid Graphs”, *IEEE Trans. Vis. Comput. Graph.*, Vol. 29(1), pp. 875–885, 2023.

URL <https://doi.org/10.1109/TVCG.2022.3209485>



■ **Figure 2** Visualizing the research groups of the Agricultural Faculty of the University of Bonn with *MosaicSets*.

Visualizing sets of elements and their relations is an important research area in information visualization. In this presentation, we present *MosaicSets*: a novel approach to create Euler-like diagrams from non-spatial set systems such that each element occupies one cell of a regular hexagonal or square grid. The main challenge is to find an assignment of the elements to the grid cells such that each set constitutes a contiguous region. As use case, we consider the research groups of a university faculty as elements, and the departments and joint research projects as sets. We aim at finding a suitable mapping between the research groups and the grid cells such that the department structure forms a base map layout. Our objectives are to optimize both the compactness of the entirety of all cells and of each set by itself. We show that computing the mapping is NP-hard. However, using integer linear programming we can solve real-world instances optimally within a few seconds. Moreover, we propose a relaxation of the contiguity requirement to visualize otherwise non-embeddable set systems. We present and discuss different rendering styles for the set overlays. Based on a case study with real-world data, our evaluation comprises quantitative measures as well as expert interviews.

3.6 (Contributed) How to visualize uncertainty

Sara Irina Fabrikant (Universität Zürich, CH)

License  Creative Commons BY 4.0 International license
© Sara Irina Fabrikant

The brief presentation introduced the interdisciplinary audience to the empirical research frontier in how to visualize uncertainty, inherent to any collected, analyzed, and visualized data. I reviewed empirically evaluated visual variables that are intuitively understood by target users of uncertainty visualizations (e.g., [1, 2]). I also reported on past and ongoing empirical geovisualization research with colleagues that investigates how data uncertainty visualized on maps might influence the process and outcomes of spatial decision-making, especially when made under time pressure, and in risky situations. Based on our collected empirical evidence to date, we argue that spatial data uncertainties should be communicated to space-time decision-makers, especially when decisions need to be made with limited time resources and when decision outcomes can have dramatic consequences.

References

- 1 MacEachren, Alan M. and Robinson, Anthony and Hopper, Susan and Gardner, Steven and Murray, Robert and Gahegan, Mark and Hetzler, Elisabeth (2005). Visualizing Geospatial Information Uncertainty: What We Know and What We Need to Know. *Cartography and Geographic Information Science*, 32(3):139–160. doi:10.1559/1523040054738936.
- 2 Alan M. MacEachren, Robert E. Roth, James O'Brien, Bonan Li, Derek Swingley, Mark Gahegan: Visual Semiotics & Uncertainty Visualization: An Empirical Study. *IEEE Trans. Vis. Comput. Graph.* 18(12): 2496-2505 (2012).

3.7 (Contributed) Visual Encodings of Temporal Uncertainty: A Comparative User Study

Silvia Miksch (TU Wien, AT)

License © Creative Commons BY 4.0 International license
© Silvia Miksch

Joint work of Theresia Gschwandtner, Markus Bögl, Paolo Federico, Silvia Miksch
Main reference Theresia Gschwandtner, Markus Bögl, Paolo Federico, Silvia Miksch: “Visual Encodings of Temporal Uncertainty: A Comparative User Study”, *IEEE Trans. Vis. Comput. Graph.*, Vol. 22(1), pp. 539–548, 2016.
URL <https://doi.org/10.1109/TVCG.2015.2467752>

Visualizing temporal uncertainty is still an open research challenge because the special characteristics of time require special visual encodings and may provoke different interpretations. Thus, we have conducted a comprehensive study comparing alternative visual encodings of intervals with uncertain start and end times: gradient plots, violin plots, accumulated probability plots, error bars, centered error bars, and ambiguation. Our results reveal significant differences in error rates and completion time for these different visualization types and different tasks. We recommend using ambiguation – using a lighter color value to represent uncertain regions – or error bars for judging durations and temporal bounds, and gradient plots – using fading color or transparency – for judging probability values.

3.8 (Contributed) Simultaneous Matrix Orderings for Graph Collections

Nathan Van Beusekom (TU Eindhoven, NL), Wouter Meulemans (TU Eindhoven, NL)

Joint work of Nathan Van Beusekom, Wouter Meulemans, and Bettina Speckmann
License © Creative Commons BY 4.0 International license
© Nathan Van Beusekom and Wouter Meulemans
Main reference Nathan van Beusekom, Wouter Meulemans, Bettina Speckmann: “Simultaneous Matrix Orderings for Graph Collections”, *CoRR*, Vol. abs/2109.12050, 2021.
URL <https://arxiv.org/abs/2109.12050>

Undirected graphs are frequently used to model phenomena that deal with interacting objects, such as social networks, brain activity and communication networks. The topology of an undirected graph G can be captured by an adjacency matrix; this matrix in turn can be visualized directly to give insight into the graph structure. Which visual patterns appear in such a matrix visualization crucially depends on the *ordering* of its rows and columns. Formally defining the quality of an ordering and then automatically computing a high-quality ordering are both challenging problems; however, effective heuristics exist and are used in practice.

Often, graphs do not exist in isolation but as part of a collection of graphs on the same set of vertices, for example, brain scans over time or of different people. To visualize such graph collections, we need a *single* ordering that works well for all matrices *simultaneously*. The current state-of-the-art solves this problem by taking a (weighted) union over all graphs and applying existing heuristics. However, this union leads to a loss of information, specifically in those parts of the graphs which are different. We propose a *collection-aware* approach to avoid this loss of information and apply it to two popular heuristic methods: leaf order and barycenter.

The de-facto standard computational quality metrics for matrix ordering capture only block-diagonal patterns (cliques). Instead, we propose to use *Moran’s I*, a spatial auto-correlation metric, which captures the full range of established patterns. Moran’s *I* refines

previously proposed stress measures. Furthermore, the popular leaf order method heuristically optimizes a similar measure which further supports the use of Moran's I in this context. An ordering that maximizes Moran's I can be computed via solutions to the Traveling Salesperson Problem (TSP); orderings that approximate the optimal ordering can be computed more efficiently, using any of the approximation algorithms for metric TSP.

We evaluated our methods for simultaneous orderings on real-world datasets using Moran's I as the quality metric. Our results show that our collection-aware approach matches or improves performance compared to the union approach, depending on the similarity of the graphs in the collection. Specifically, our Moran's I -based collection-aware leaf order implementation consistently outperforms other implementations. Our collection-aware implementations carry no significant additional computational costs.

3.9 (Contributed) On Full Diversity in Metric Spaces

Marc van Kreveld (Utrecht University, NL)

License © Creative Commons BY 4.0 International license
© Marc van Kreveld

Joint work of Fabian Klute, Marc van Kreveld

Main reference Fabian Klute, Marc J. van Kreveld: "On Fully Diverse Sets of Geometric Objects and Graphs", in Proc. of the Graph-Theoretic Concepts in Computer Science – 48th International Workshop, WG 2022, Tübingen, Germany, June 22-24, 2022, Revised Selected Papers, Lecture Notes in Computer Science, Vol. 13453, pp. 328–341, Springer, 2022.

URL https://doi.org/10.1007/978-3-031-15914-5_24

In a metric space we have objects and a way to measure distances between pairs of objects. In a bounded metric space, there is an upper bound on the maximum distance.

We define full diversity of a subset of a metric space as a subset where all pairs of objects are approximately as far apart as the diameter, up to a constant factor.

We examine how large fully diverse subsets can be in several cases of metric spaces, like bit strings with Hamming distance, graphs with edit distance, simple polygons inside a unit square with area-of-symmetric difference or Hausdorff distance, or Frechet distance of the boundary. We give upper and lower bounds in these cases and others.

This research is joint work with Fabian Klute and it appeared in WG 2022.

3.10 (Contributed) StoryLines

Alexander Wolff (Universität Würzburg, DE)

License © Creative Commons BY 4.0 International license
© Alexander Wolff

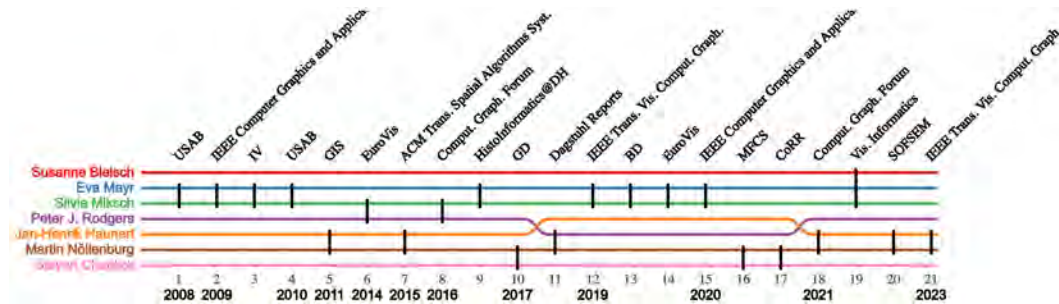
Joint work of Alexander Wolff, Tim Herrmann

In this talk I sketch the main idea of a master's thesis [1] that a student, Tim Herrmann, wrote in my group recently. He developed a web service called *PubLines*¹ where the user can enter a few names of computer scientists. Then Tim Herrmann's JavaScript program scans the computer science bibliography *dblp*² for publications that are co-authored by at least two

¹ <https://www1.pub.informatik.uni-wuerzburg.de/pub/publines>

² <https://dblp.uni-trier.de>

of the names that the user entered. The program outputs a so-called *storyline visualization* of the sequence of joint publications. In this visualization, the authors are represented by x-monotone curves and their joint publications are represented by vertical bars such that the curve of a given author goes through the bars of exactly those publications that (s)he has co-authored. The program heuristically tries to minimize the (block) crossings [2] between the curves. For an example that visualizes the joint publications of some seminar participants, see Figure 3.



■ **Figure 3** A storyline visualization of the joint publications of some seminar participants.

References

- 1 T. Herrmann. Storyline visualizations for scientific collaboration graphs. Master’s thesis, Institut für Informatik, Universität Würzburg, 2022. URL: <https://www1.pub.informatik.uni-wuerzburg.de/pub/theses/2022-herrmann-masterarbeit.pdf>.
- 2 T. C. van Dijk, M. Fink, N. Fischer, F. Lipp, P. Markfelder, A. Ravsky, S. Suri, and A. Wolff. Block crossings in storyline visualizations. *J. Graph Alg. Appl.*, 21(5):873–913, 2017. doi:10.7155/jgaa.00443.

4 Working groups

4.1 StorySets

Annika Bonerath (Universität Bonn, DE), Stephen G. Kobourov (University of Arizona – Tucson, US), Wouter Meulemans (TU Eindhoven, NL), Martin Nöllenburg (TU Wien, AT), Markus Wallinger (TU Wien, AT), and Alexander Wolff (Universität Würzburg, DE)

License © Creative Commons BY 4.0 International license
 © Annika Bonerath, Stephen G. Kobourov, Wouter Meulemans, Martin Nöllenburg, Markus Wallinger, and Alexander Wolff

Problem Definition

Roughly speaking, there are two types of set systems: abstract and spatial. In a spatial set system, each element has a fixed, spatial location. Visualizations of spatial set systems usually use connected regions of the plane in order to visualize the sets: elements are points in the plane, and a set consists of exactly those elements that lie inside the region. Examples for such visualizations are Bubble Sets [4], GMap [8], or MapSets [6]. LineSets [1] connect the elements of a set by a path. LineSets are generalized by Kelp diagrams [5], which connect the elements of a set by a sparse spanning graph. In order to avoid intersections, ClusterSets [9] allow disconnected regions to visualize sets. Examples for visualizations of abstract set systems are Venn diagrams, Euler diagrams [7], linear diagrams [11], matrices, etc. The survey by Alsallakh et al. [2] gives an extensive overview over methods for set visualization.

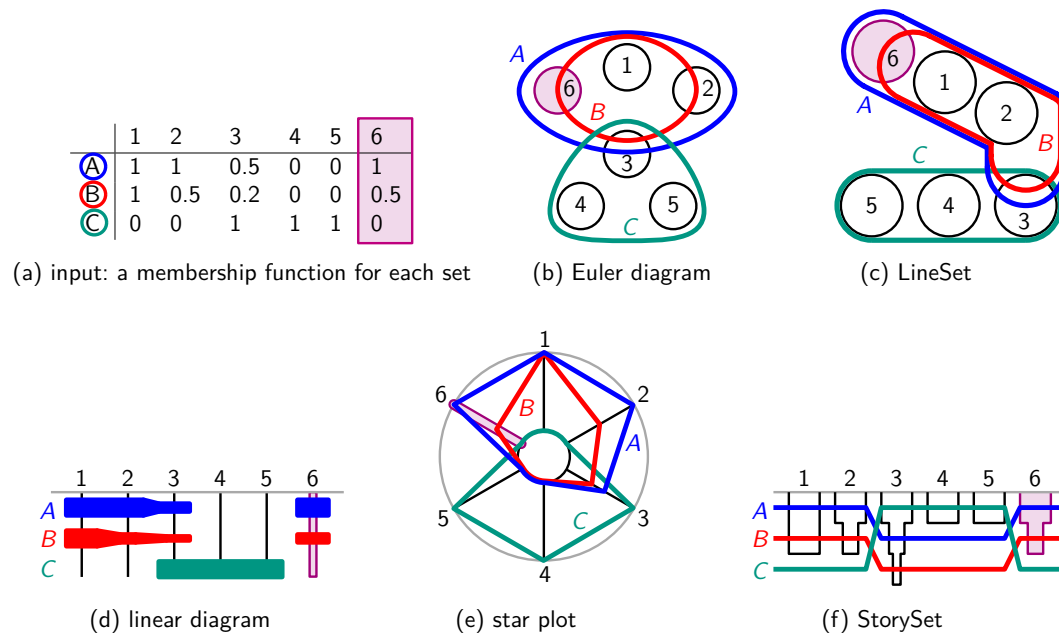


Figure 4 Various graphical representations of a fuzzy set system with three sets A, B, C and six elements 1, 2, 3, 4, 5, 6. The table in (a) lists the uncertain set memberships.

A fuzzy set system (U, \mathcal{F}) consists of a finite universe U of elements and a finite collection \mathcal{F} of fuzzy sets. A fuzzy set S in \mathcal{F} is described by a membership function $\mu_S: U \rightarrow [0, 1]$ that expresses the certainty of element x being in set S . For example, if $\mu_S(x) = 1$, then x is certainly contained in S , and if $\mu_S(x) = 0$, then x is certainly not contained in S . We assume that membership functions are independent of each other.

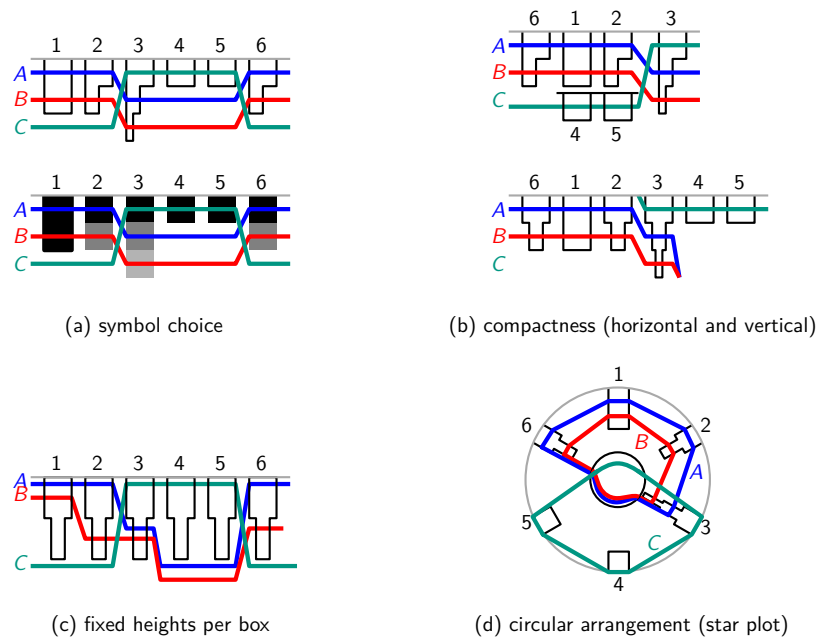
During the seminar, we investigated various ways to graphically represent fuzzy set systems. We focussed on visualizing (uncertain) set membership per element and (uncertain) set containment. Specifically, if $\mu_S(x) \leq \mu_{S'}(x)$ for every $x \in U$, then we say that S is *uncertainly contained* in S' , and we would like this relationship to be graphically represented as well. Note, however, that uncertain containment does not guarantee containment.

Exploring the Design Space

During the seminar, we first tried to extend visualizations for spatial set systems to the fuzzy case. We considered representing elements as unit-sized disks instead of points; see Figure 4b for an adaption of Euler diagrams and Figure 4c for an adaption of LineSets. In order to indicate how certain the membership of an element in a set is we used the element’s distance to the set boundary or partial containment of the element in the set’s region. The latter type of representation may be somewhat more intuitive; see Figure 4b and Figure 4c.

Second, we looked at an adaptation of linear diagrams [11]; see Figure 4d. Each set is a row in this matrix, and each element is a column. The certainty of the membership is illustrated by the line width. A disadvantage of this visualization is that it is difficult to recognize (uncertain) set containment.

Third, we explored a style in which sets are represented as curves and elements as axes that are intersected by the set curves. For example, in a star plot [3], axes are radials emanating from a common center and a set (or, by duality, an element) is represented by a polygon whose vertices lie on these axes. The distance of a polygon vertex from the center of



■ **Figure 5** Different design variants of StorySets.

the diagram expresses the value of some parameter. For visualizing fuzzy set systems, we introduced an axis for each element x and chose as parameter to be displayed on its axis the membership certainty $\mu_S(x)$ of x in S ; see Figure 4e. This visualization easily encodes uncertain set containment via polygon containment.

However, this principle can be applied more generally, using ideas found in, e.g., Sankey diagrams [12] and storyline visualizations [10, 13]. Specifically, we explored a novel design that combines storylines and set visualization. For this design, we assume that membership certainties are grouped into a few *certainty groups*, e.g., by rounding. Now, we can partition the axis of an element according to the number of sets per uncertainty group. We represent every element by a stack of boxes such that each box represents one certainty group. In this representation, the width of a box encodes the certainty and the height of a box encoding the number of sets that contain the element with the given certainty. The box for the highest certainty group is anchored at the top and the others are placed underneath, ordered by their certainty value. We call this visualization *StorySets*; see Figure 4f. With this visualization (uncertain) set containment is encoded as follows: if the curve of a set S is always below the curve of another set S' , then the S is uncertainly contained in S' . We consider StorySets a promising approach for the visualization of fuzzy set systems and, hence, focused on StorySets for the remainder of the seminar.

During the discussion, we considered several variants of StorySets, some needing only small adjustments (e.g., a different choice of symbols; see Figure 5a); others posing new algorithmic problems (e.g., compactness; see Figure 5b). In the following, we discuss three of these variants.

Compactness

We can vertically compact a StorySet representation by not showing every set curve for the whole time, that is, over the complete horizontal extent of the diagram. Instead, we allow each set curve to start immediately before its respective first element in the given element order and to end right after its respective last element; see Figure 5b.

Also, it is possible to horizontally compact the design by considering “non-linear” storylines. To this end, we can stack an element x above an element y if (it is certain that) there is no set that contains both x and y ; see Figure 5b.

Fixed heights per certainty box

Due to the varying heights of the certainty boxes throughout the elements, it might be difficult to perceive whether a curve changes the certainty level between two elements or not. Hence, we considered the variant that we have a fixed height per certainty group defined by the maximal number of sets in which an element is contained; see Figure 5c.

Circular arrangement

We want to mention that the optimization for horizontal arrangement of elements can be easily adapted to a circular arrangement of elements. Here, one only needs to take care about the additional cyclic constraints between the last and the first element.

Algorithmic Considerations

The designs discussed in the last section give rise to several optimization problems. In the case of star plots, linear diagrams or StorySets, the order of the elements is very important. Ideally, we would like to place elements with similar fuzzy set memberships next to each other resulting in more homogeneous curves and polygons. Since each element can be seen as an $|\mathcal{F}|$ -dimensional vector in a matrix, we can define a distance metric between them and model this as an ordering problem. Depending on the visualization design, computing the linear arrangement or profile of the matrix, or modelling it as a traveling salesperson problem, could be used to tackle this optimization problem.

Specifically for StorySets, it seems natural to consider the following optimization goals: (i) minimize the number of curve crossings, (ii) minimize the number of level changes over all curves, and (iii) keep the drawing compact, that is, minimize width and/or height. In order to achieve these design goals, we want to find a horizontal order of the elements and a vertical order of the curves that respects the certainty groups of each element. Note that goals (i) and (ii) are very similar to what standard storylines aim to optimize [10].

We worked on a pipeline that alternates between the optimization of the horizontal order of the elements and the vertical order of the curves per certainty box and per element. First, we fix the vertical order of curves per element such that the certainty groups are respected. Second, we compute the number of crossings between every pair of elements. Third, we introduce an auxiliary complete graph where each vertex corresponds to an element and each edge is weighted with the number of crossing curves if the two elements are consecutive in the horizontal order. Then, we solve the traveling salesperson problem on this graph. This leads to a new horizontal order of the elements. Now, if we can further improve the vertical order of the curves per element, we return to the first step. Otherwise, the algorithm terminates.

Outlook


We plan to provide an framework with several algorithms for computing various versions of StorySets. We want to evaluate our approach using real-world data. Such real-world data could be, for example, a set of tweets combined with a topic analysis. Each topic then corresponds to a set and the certainty with which a tweet belongs to a topic reflects its uncertain membership in that set.

References

- 1 B. Alper, N. Henry Riche, G. Ramos, and M. Czerwinski. Design study of LineSets, a novel set visualization technique. *IEEE Trans. Vis. Comput. Graphics*, 17(12):2259–2267, 2011. doi:10.1109/TVCG.2011.186.
- 2 B. Alsallakh, L. Micallef, W. Aigner, H. Hauser, S. Miksch, and P. J. Rodgers. The state-of-the-art of set visualization. *Comput. Graphics Forum*, 35(1):234–260, 2016. doi:10.1111/cgf.12722.
- 3 J. Chambers, W. Cleveland, B. Kleiner, and P. Tukey. *Graphical Methods for Data Analysis*. Wadsworth, 1983.
- 4 C. Collins, G. Penn, and S. Carpendale. Bubble Sets: Revealing set relations with isocontours over existing visualizations. *IEEE Trans. Vis. Comput. Graphics*, 15(6):1009–1016, 2009. doi:10.1109/TVCG.2009.122.
- 5 K. Dinkla, M. van Kreveld, B. Speckmann, and M. A. Westenberg. Kelp diagrams: Point set membership visualization. *Comput. Graphics Forum*, 31(3):875–884, 2012. doi:10.1111/j.1467-8659.2012.03080.x.
- 6 A. Efrat, Y. Hu, S. G. Kobourov, and S. Pupyrev. MapSets: Visualizing embedded and clustered graphs. *J. Graph Algorithms Appl.*, 19(2):571–593, 2015. doi:10.7155/jgaa.00364.
- 7 J. Flower, A. Fish, and J. Howse. Euler diagram generation. *J. Vis. Lang. Comput.*, 19(6):675–694, 2008. doi:10.1016/j.jvlc.2008.01.004.
- 8 E. R. Gansner, Y. Hu, and S. G. Kobourov. GMap: Visualizing graphs and clusters as maps. In *Proc. Pacific Vis. Symp. (PacificVis)*, pages 201–208. IEEE, 2010. doi:10.1109/PACIFICVIS.2010.5429590.
- 9 J. Geiger, S. Cornelsen, J.-H. Haunert, P. Kindermann, T. Mchedlidze, M. Nöllenburg, Y. Okamoto, and A. Wolff. ClusterSets: Optimizing planar clusters in categorical point data. *Comput. Graphics Forum*, 40(3):471–481, 2021. doi:10.1111/cgf.14322.
- 10 I. Kostitsyna, M. Nöllenburg, V. Polishchuk, A. Schulz, and D. Strash. On minimizing crossings in storyline visualizations. In E. D. Giacomo and A. Lubiw, editors, *Proc. 23rd Int. Symp. Graph Drawing Network Vis. (GD’15)*, volume 9411 of *LNCS*, pages 192–198. Springer, 2015. URL: <https://arxiv.org/abs/1509.00442>, doi:10.1007/978-3-319-27261-0_16.
- 11 P. J. Rodgers, G. Stapleton, and P. Chapman. Visualizing sets with linear diagrams. *ACM Trans. Comput. Hum. Interact.*, 22(6):27:1–27:39, 2015. doi:10.1145/2810012.
- 12 M. H. P. R. Sankey. Introductory note on the thermal efficiency of steam-engines. In *Minutes of Proceedings of the Institution of Civil Engineers*, volume 134, pages 278–283, 1898.
- 13 T. C. van Dijk, M. Fink, N. Fischer, F. Lipp, P. Markfelder, A. Ravsky, S. Suri, and A. Wolff. Block crossings in storyline visualizations. *J. Graph Alg. Appl.*, 21(5):873–913, 2017. doi:10.7155/jgaa.00443.

4.2 Simplification of Set Systems to Draw Nice Euler Diagrams

Peter Rodgers (University of Kent – Canterbury, GB), Daniel Archambault (Swansea University, GB), Jan-Henrik Haunert (Universität Bonn, DE), and Bei Wang Phillips (University of Utah – Salt Lake City, US)

License  Creative Commons BY 4.0 International license

© Peter Rodgers, Daniel Archambault, Jan-Henrik Haunert, and Bei Wang Phillips

Motivation and general idea

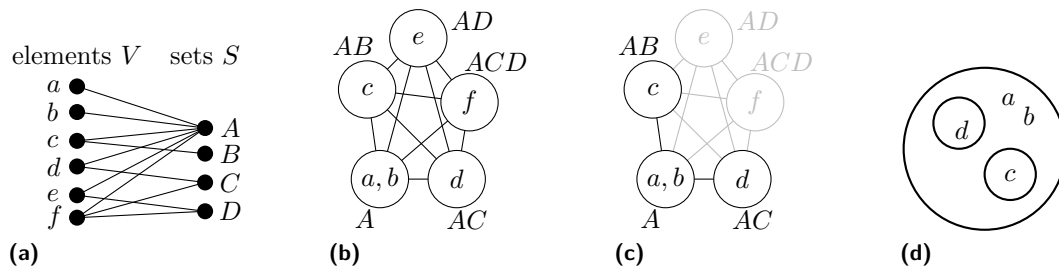
Euler diagrams are frequently used to visualize set systems. They represent each set as a region that is bounded by a closed curve. An area in an Euler diagram where multiple regions overlap indicates the existence of set elements that are contained in the corresponding sets and in no other set. An advantage of Euler diagrams is that they are intuitive to understand. However, they can become cluttered even for medium-sized set systems. When requiring the regions to be contiguous, it may even be the case that for a given set system no Euler diagram exists. We aim to mitigate these weaknesses of Euler diagrams by developing algorithms for the simplification of set systems. Given a set system as input, our goal is to compute a simplified but still similar version of it that can be drawn nicely as an Euler diagram. In this report we introduce a measure of dissimilarity between a set system and a simplified version of it (i.e., a distance) and state the criteria a set system needs to fulfill such that it can be drawn nicely. Based on this we sketch an optimization approach as well as a greedy heuristic. While a lot of previous work on the automatic generation of Euler diagrams exists [1, 2, 3, 4, 5, 6, 7, 8], we hope that with our simplification algorithms we will be able to extend the applicability of Euler diagrams to much larger and more complex set systems.

Preliminaries

Formally we consider a set system as a bipartite graph $\mathcal{S} = (V, S; E)$; see Fig. 6a. Every node $u \in V$ represents an element and every node $X \in S$ a set. Every edge $\{u, X\} \in E$ indicates that element $u \in V$ is a member of set $X \in S$. The set of all elements S is partitioned into *zones* such that, if two elements $u, v \in V$ are members of exactly the same sets in S , then u and v are in the same zone. A zone is *labeled* with the sets in S that have its elements as members. With $\ell(Z)$ we refer to the set of labels of a zone Z . Our algorithms operate on a graph \mathcal{G} that contains a node for each zone and an edge between every two zones that share at least one label; see Fig. 6b. In particular, our optimization algorithm computes a subgraph \mathcal{G}' of \mathcal{G} such that, in a next processing step, we can compute an Euler diagram whose dual graph is \mathcal{G}' . Accordingly, we call \mathcal{G} the *super dual graph*. Figure 6c shows a subgraph \mathcal{G}' of the super dual graph \mathcal{G} in Fig. 6b. An Euler diagram that has \mathcal{G}' as its dual graph is shown in Fig. 6d.

Distance measure

Since we aim for a set system that is similar to the input set system \mathcal{S} , we need to quantify the loss of information resulting from the reduction of the super dual graph \mathcal{G} to the dual graph \mathcal{G}' of the Euler diagram. We do this by computing an optimal assignment that maps every zone that is not selected for \mathcal{G}' to a selected zone or to the empty set \emptyset . Each assignment of a zone Z_1 to a zone Z_2 is interpreted such that the elements in Z_1 are removed from every set in $\ell(Z_1) \setminus \ell(Z_2)$ and inserted into every set in $\ell(Z_2) \setminus \ell(Z_1)$. If Z_1 is assigned to \emptyset , then the elements in Z_1 are removed from every set in $\ell(Z_1)$ but not inserted into any set. For



■ **Figure 6** From left to right: (a) A bipartite graph \mathcal{S} representing a set system, (b) the super dual graph \mathcal{G} for \mathcal{S} , (c) the subgraph \mathcal{G}' of \mathcal{G} that has been computed to form the dual graph of the Euler diagram, (d) the Euler diagram.

every removal of an element from a set and every insertion of an element into a set we charge a cost $c_{\text{remove}} \in \mathbb{R}_{\geq 0}$ and $c_{\text{insert}} \in \mathbb{R}_{\geq 0}$, respectively. The *transport distance* between the two set systems represented by \mathcal{G} and \mathcal{G}' is the minimum total cost over all possible assignments.

Optimization problem

We are now ready to define the task of simplifying a set system as an optimization problem. **SETSYSTEMSIMPLIFICATION:** Given the super dual graph \mathcal{G} of a set system \mathcal{S} , find a subgraph \mathcal{G}' of \mathcal{G} such that

- (1) \mathcal{G}' is planar,
- (2) for every set $X \in S$, the subgraph of \mathcal{G}' induced by the set of zones labeled with X is connected or empty, and
- (3) the transport distance between the set systems represented by \mathcal{G} and \mathcal{G}' is minimized.

Optionally, to keep the visual complexity of the output Euler diagram low, we could require to select at most a prescribed number k of zones. Furthermore, to avoid concurrencies in the output Euler diagram (which is one of the criteria for well-formed Euler diagrams), we could discard all edges of \mathcal{G} whose incident zones differ by more than one label. To compute an optimal solution, we consider Integer Linear Programming, but also local search seems promising.

Greedy heuristic

Our greedy heuristic takes the super dual graph \mathcal{G} as input and iterates until a certain set of conditions is reached. In each iteration, we select two intersecting sets A and B and merge them to one. We update \mathcal{G} to reflect this change. In principle, we could choose A and B arbitrarily (as long as they intersect). However, to get closer to a sufficiently simplified version of the set system, we prioritize the different options as follow:

- A merge of two sets A and B has highest priority if it destroys a subgraph of \mathcal{G} that is a Kuratowski subdivision, i.e., a subdivision of one of the two non-planar graphs K_5 or $K_{3,3}$. This is because it will bring us closer to a planar graph, which is strictly required for drawing the Euler diagram.
- Our second priority is to choose a merge if it destroys a concurrency.
- If multiple merges are equally good according to the first two criteria, we choose the merge based on a measure of (dis-)similarity. In particular, we consider choosing the merge that yields the set system at minimum transport distance from the current set system, but it would also be reasonable to merge the two sets with maximum Jaccard similarity.

Final remarks

We consider the optimization approach and the greedy heuristic as two complementary approaches, which we plan to implement and compare in experiments. Concerning the rendering of an Euler diagram based on its dual graph our hope is that we can re-use existing methods and software. A crucial question that needs further discussions, however, is how the uncertainties introduced with our simplification methods can be visualized.

References

- 1 Jean Flower and John Howse. Generating Euler diagrams. In *Proc. 2nd International Conference on Diagrammatic Representation and Inference (Diagrams 2002)*, pages 61–75, 2002.
- 2 Peter Rodgers, Leishi Zhang, and Andrew Fish. General Euler diagram generation. In *Proc. 5th International Conference on Diagrammatic Representation and Inference (Diagrams 2008)*, pages 13–27, 2008.
- 3 Peter Rottmann, Markus Wallinger, Annika Bonerath, Sven Gedicke, Martin Nöllenburg, and Jan-Henrik Haurert. MosaicSets: Embedding set systems into grid graphs. *IEEE Transactions on Visualisation and Computer Graphics*, 29(1):875–885, 2023.
- 4 Paolo Simonetto, Daniel Archambault, and Carlos Scheidegger. A simple approach for boundary improvement of Euler diagrams. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):678–687, 2016.
- 5 Paolo Simonetto and David Auber. Visualise undrawable Euler diagrams. In *Proc. 12th International Conference Information Visualisation*, pages 594–599, 2008.
- 6 Paolo Simonetto and David Auber. An heuristic for the construction of intersection graphs. In *Proc. 13th International Conference Information Visualisation*, pages 673–678, 2009.
- 7 Paolo Simonetto, David Auber, and Daniel Archambault. Fully automatic visualisation of overlapping sets. In *Proc. 11th Eurographics / IEEE – VGTC Symposium on Visualization (EuroVis’09)*, pages 967–974, 2009.
- 8 Youjia Zhou, Archit Rathore, Emilie Purvine, and Bei Wang. Topological simplifications of hypergraphs. *IEEE Transactions on Visualization and Computer Graphics*, 2022.

4.3 Types of Uncertainty in Set Visualization

Christian Tominski (Universität Rostock, DE), Michael Behrisch (Utrecht University, NL), Susanne Bleisch (FH Nordwestschweiz – Muttenz, CH), Sara Irina Fabrikant (Universität Zürich, CH), Eva Mayr (Donau-Universität Krems, AT), Silvia Miksch (TU Wien, AT), and Helen C. Purchase (Monash University – Clayton, AU)

License © Creative Commons BY 4.0 International license

© Christian Tominski, Michael Behrisch, Susanne Bleisch, Sara Irina Fabrikant, Eva Mayr, Silvia Miksch, and Helen C. Purchase

Set visualization deals with visual methods to support people understand and make sense of sets, their elements, and relations thereof. Existing methods such as Euler diagrams, Venn diagrams, and bi-partite node-link representations focus on communicating set memberships, their cardinality, and their possible intersections. However, designing visual representations of uncertain sets appears to be challenging. This is mainly due to the fact that not only the data D themselves need to be encoded visually, but also the information about their

uncertainty U needs to be communicated to a reader. Above all, set visualization users must be able to extract all the encoded information (about the data and their uncertainty) from the visualization, which can be formulated abstractly as a pipeline:

$$(D, U) \xrightarrow{m} V \xrightarrow{i} (D', U').$$

The visualization designer defines a mapping m of data D and uncertainty U to create a visual representation V . Through an interpretation i of the visual representation V , human observers extract their own versions of data D' and uncertainty information U' . The scientific challenge is to understand the cognitive process of i and to devise mappings m so that ideally $D = D'$ and $U = U'$ for all human observers. The congruence of D and D' , as well as U and U' , can serve as a guiding principle for the visualization of uncertain data.

While set visualizations themselves are an active research frontier there are far fewer research activities in the understanding of the implications of uncertainty for set visualization. In the first place, it is still unclear how uncertainty is defined in the context of set-type data. Only if we know what types of uncertainty are relevant for set type data can we design expressive visual representations of uncertain sets. Therefore, we conceptualized uncertainty in the context of set visualization by examining (a) which aspects of set-type data might be affected by uncertainty, and (b) which characteristics of uncertainty might influence the visualization design.

Undeniably, uncertainty bears the notion of something being known, unknown, vague, and/or containing varying accuracy. So, the starting point of our discussion centered around specifying what is known and what is unknown. In a perfect world, we know the data and we assume that they are accurate. For set-type data this means that we know for certain all elements, all existing sets, and the set membership of each element. There are also associated data attributes we know with certainty, for example, set size as an important derived set attribute. There may be further data attributes given for elements or sets for which we know their data values with certainty (e.g., the number of female members of a team). Given these data characteristics (D), the visualization of set-type data is primarily concerned with communicating (i) set membership, (ii) set properties, and (iii) associated data attributes. An overview of suitable visualization methods for the cases where set characteristics are certain is available in [1].

While we might believe to know things accurately in a perfect world, in the real world, however, there is uncertainty about uncertainty (U) surrounding us. Just take the weather predictions, for example, and the often heard statement “There is a 70% chance of rain tomorrow” on your favorite weather app. We thus asked ourselves, how much do we actually know about data uncertainty? In a perfect world, we know that there exists no uncertainty at all, which we denote as $U = 0$. In the real world, however, one can distinguish two scenarios. First, we know that there is uncertainty, but we cannot tell accurately where it is, what it is, or how much of it exists. In other words, we know for a fact that uncertainty is present in our data, but no further details. We denote this as $U > 0$. In the second scenario, we not only know that uncertainty exists in our data, but we also know with certainty where, what, and how much of it exists in our data. For the sake of simplicity, we denote this as $U = p$. The letter p is a strong simplification of what could be known about the uncertainty in our dataset. Depending on the given data characteristics we are interested in, p can take different forms. When set membership of an element a and a set X is certain, one can say either $a \in X$ or $a \notin X$. Under uncertainty, p might denote a probability of a being a member of X , $P(a, X) = p$, which is a notation known from fuzzy sets. In this case, p can be understood as a plain probability value. Yet, we could also say that p denotes a more complex probability

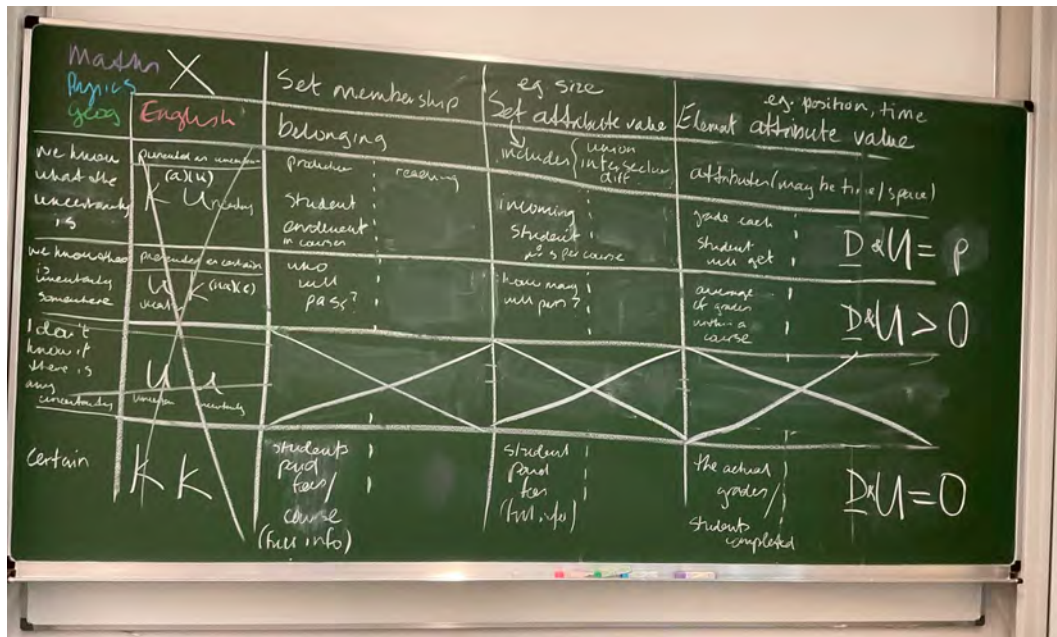


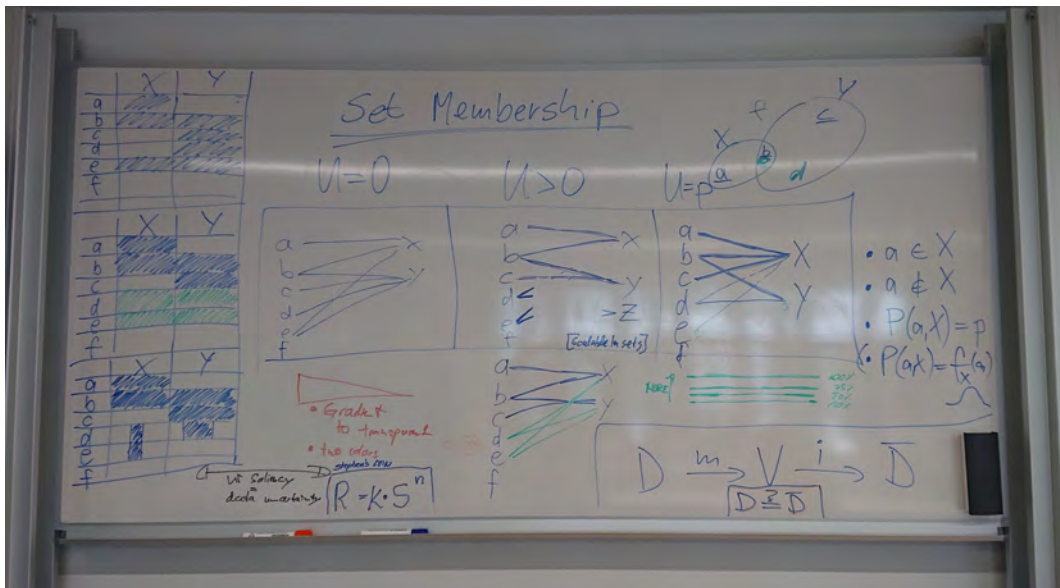
Figure 7 Conceptual framework sketched as a table with columns and rows representing data characteristics D and types of uncertainty U , respectively. The individual columns were discussed in separate subgroups.

distribution (e.g., $p = \mathcal{N}(\mu, \sigma^2)$) based on which set membership is decided. Also, in relation to the data attributes of elements or sets, we may understand p as the probability value of an attribute taking a particular data value. The same holds for the notion of p being a probability distribution. Additionally, it is common for uncertain attribute values to specify them via a range of possible values, in which case $p = [l, u]$ is some interval with a lower and upper bound of l and u .

Overall, the discussion of the characteristics of set data D and the types of uncertainty U led us to a conceptual framework of uncertainty in set visualization. In terms of D , the framework distinguishes: set membership, set attributes, and element attributes. Related to U , we use the different plausible types of (un)certainly: certainty ($U = 0$), uncertainty as a binary fact ($U > 0$), and uncertainty as quantifiable measure ($U = p$). We captured the framework in a table whose columns and rows respectively represent D and U , as shown in fig. 7.

Based on this conceptual framework, we then systematically discussed possible visualization designs to illustrate examples and highlight challenges of integrating uncertainty in set visualizations. Three subgroups were formed, each working on a selected data characteristic (i.e., table column). As a baseline, each subgroup used the simple case of a visual representation with zero uncertainty ($U = 0$). The group that dealt with set membership worked with bi-partite node-link and matrix representations, which were gradually expanded to include unknown uncertainty ($U > 0$) and known uncertainty ($U = p$) by varying the visual encoding of links and matrix cells as indicated in fig. 8.

Set attributes turned out to be particularly challenging to visualize when uncertainty is involved. The reason for this is that derived data attributes depend by definition on other data characteristics, which also can include varying levels of uncertainty. For example, set size depends on set memberships. Leaving this particular challenge for future work, the

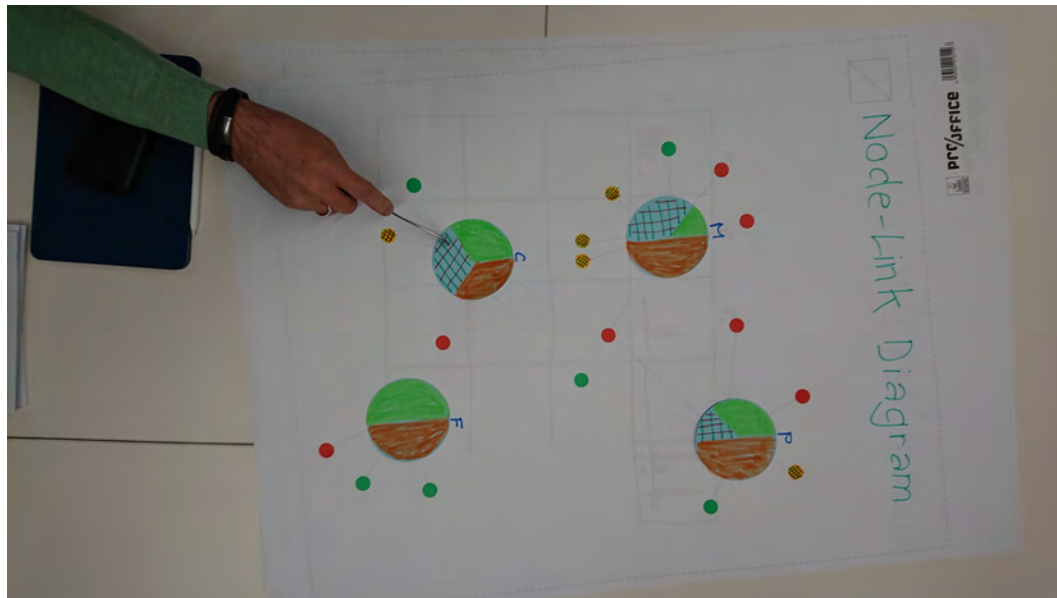


■ **Figure 8** Integrating uncertainty into visual representations of set membership. Two alternatives were sketched: bi-partite node-link representation with uncertainty encoded on the links (center) and matrix visualization with uncertainty visualized in the matrix cells.

subgroup designed and discussed visual representations where set attributes do not depend on other factors. They came up with node-link-style representations as shown in fig. 9. Sets are represented as bigger nodes being linked to their belonging set elements, which are depicted as smaller nodes. The set attributes are shown as pie charts within the bigger nodes, where color hue indicates certain attribute values and hatching marks uncertain set elements. The same encoding is applied to the attributes of the individual set elements on the smaller nodes.

Finally, one subgroup worked on visualizing uncertain element attributes. Their focus was not so much on coming up with new designs, but to review the existing knowledge about general uncertainty visualization. For example, cartography has a long history in working with uncertain data, but also the visualization community studied this topic in detail. Particularly, the works by Alan MacEachren et al. [5, 6], Kristin Potter et al. [7, 8, 2], Amit Jena et al. [4], and Theresia Gschwandtner et al. [3] offer profound insight into how uncertain data values can be encoded visually, and to what degree humans can interpret and understand the depicted information. With these general considerations, the table of the developed conceptual framework could be filled completely. Based on the intense and productive discussions centered on the conceptual framework for set visualization and uncertainty, we drafted an outline for a journal article that will summarize key results of the research conducted at the Dagstuhl-Seminar. Our planned article will also include a synthesis of recommendations to be considered when designing visualizations for uncertain set data and an outline of future research directions.

This working group consisted of (in alphabetical order) Michael Behrisch, Susanne Bleisch, Sarah Fabrikant, Eva Mayr, Silvia Miksch, Helen Purchase, and Christian Tominski (see fig. 10). Helen Purchase headed the group. Christian Tominski drafted this report. All members of the team contributed significantly to the discussions, provided feedback and edited this report, and will be co-authors of the planned journal article.



■ **Figure 9** Node-link depiction where larger nodes visualize attributes of sets by color hue (certain) or by hatching (uncertain) within a pie chart and smaller nodes denote set elements using the same visual variables.



■ **Figure 10** Members of the working group (from left to right): Helen Purchase (lead), Susanne Bleisch, Christian Tominski, Eva Mayr, Silvia Miksch, Sarah Fabrikant, and Michael Behrisch.

References

- 1 Bilal Alsallakh, Luana Micalef, Wolfgang Aigner, Helwig Hauser, Silvia Miksch, and Peter J. Rodgers. The State-of-the-Art of Set Visualization. *Comput. Graph. Forum*, 35(1):234–260, 2016. doi:10.1111/cgf.12722.
- 2 Georges-Pierre Bonneau, Hans-Christian Hege, Chris R. Johnson, Manuel M. Oliveira, Kristin Potter, Penny Rheingans, and Thomas Schultz. Overview and State-of-the-Art of Uncertainty Visualization. In Charles D. Hansen, Min Chen, Christopher R. Johnson, Arie E. Kaufman, and Hans Hagen, editors, *Scientific Visualization, Mathematics and Visualization*, pages 3–27. Springer, 2014. doi:10.1007/978-1-4471-6497-5_1.
- 3 Theresia Gschwandtner, Markus Bögl, Paolo Federico, and Silvia Miksch. Visual Encodings of Temporal Uncertainty: A Comparative User Study. *IEEE Trans. Vis. Comput. Graph.*, 22(1):539–548, 2016. doi:10.1109/TVCG.2015.2467752.
- 4 Amit Jena, Ulrich Engelke, Tim Dwyer, Venkatesh Raiamanickam, and Cécile Paris. Uncertainty Visualisation: An Interactive Visual Survey. In *2020 IEEE Pacific Visualization Symposium, PacificVis 2020, Tianjin, China, June 3-5, 2020*, pages 201–205. IEEE, 2020. doi:10.1109/PacificVis48177.2020.1014.
- 5 Alan M. MacEachren, Anthony Robinson, Susan Hopper, Steven Gardner, Robert Murray, Mark Gahegan, and Elisabeth Hetzler. Visualizing Geospatial Information Uncertainty: What We Know and What We Need to Know. *Cartography and Geographic Information Science*, 32(3):139–160, 2005. doi:10.1559/1523040054738936.
- 6 Alan M. MacEachren, Robert E. Roth, James O’Brien, Bonan Li, Derek Swingley, and Mark Gahegan. Visual Semiotics & Uncertainty Visualization: An Empirical Study. *IEEE Trans. Vis. Comput. Graph.*, 18(12):2496–2505, 2012. doi:10.1109/TVCG.2012.279.
- 7 Kristin Potter. *The Visualization of Uncertainty*. PhD thesis, University of Utah, USA, 2010.
- 8 Kristin Potter, Joe Kniss, Richard F. Riesenfeld, and Chris R. Johnson. Visualizing Summary Statistics and Uncertainty. *Comput. Graph. Forum*, 29(3):823–832, 2010. doi:10.1111/j.1467-8659.2009.01677.x.

4.4 Set Size Visualization with Dependent and Independent Uncertainties

Nathan Van Beusekom (TU Eindhoven, NL), Steven Chaplick (Maastricht University, NL), Amy Griffin (RMIT University – Melbourne, AU), Martin Krzywinski (BC Cancer Research Centre – Vancouver, CA), Marc van Kreveld (Utrecht University, NL), and Hsiang-Yun Wu (FH – St. Pölten, AT)

License © Creative Commons BY 4.0 International license

© Nathan Van Beusekom, Steven Chaplick, Amy Griffin, Martin Krzywinski, Marc van Kreveld, and Hsiang-Yun Wu

Problem Setting

We discuss the visualization of set sizes and their uncertainties. We distinguish between independent and dependent uncertainty in set sizes, where the latter refers to elements that are certainly present, but it is unknown to which set they belong, among two (or more) possibilities. We present three options to visualize sets sizes and their uncertainties. For each, we discuss when to use them and what their advantages and disadvantages are.

Related Work

Sets are models that have been used in data management and analysis to capture collection relationships of elements. In addition, uncertainty information provides more context regarding the reliability of the underlying data sets. Classical uncertainty visualizations cover not only visual language design [6], but also its corresponding visual efficiency [4]. Although several set visualization algorithms have been proposed [1], integrating uncertainty into these approaches is still in its infancy.

Visualizing uncertainty in sets is related to the problem of visualizing Fuzzy Sets [10], where membership is a value between 0 and 1. Disk diagrams [8] interactively visualize fuzzy sets, and show the membership distribution of the elements with one disk per set.

Some visualization approaches for independent set uncertainty have been proposed. For example, Uncertainty Treemaps [9], introduced nested hatched lines to show the independent set uncertainty for treemaps across hierarchies. Another example of visualizing independent set size uncertainty for hierarchical data is Bubble Treemaps [3], which use squiggly lines to indicate uncertainty. UpSet [5] lists essential combinations of sets, especially their intersections, and aggregates of intersections. Showing independent and dependent set size uncertainties has not yet been fully investigated to the best of our knowledge.

Visual Design for Set Data with Dependent and Independent Uncertainties

The visualization of set sizes rather than sets with their elements allows us to use part of the available space for visualizing the set size uncertainty, since we do not need to show the elements in the sets themselves. Visualization of set size is best done by using a visual variable related to size, e.g., area or length. This holds true for the uncertainty in the set size as well. Studies show that people can intuitively understand uncertainties when they are expressed as frequencies [2]. Directly depicting both set sizes and set size uncertainties with size makes comparisons between sets easier by offloading the cognitive effort of mental arithmetic onto vision [7].

There are various scenarios where understanding the size of a set is more important than understanding set membership. Some of these scenarios have a geographic component while others do not. We therefore consider different visualization options.

A visualization of set sizes and their uncertainties should be able to show answers to the following questions:

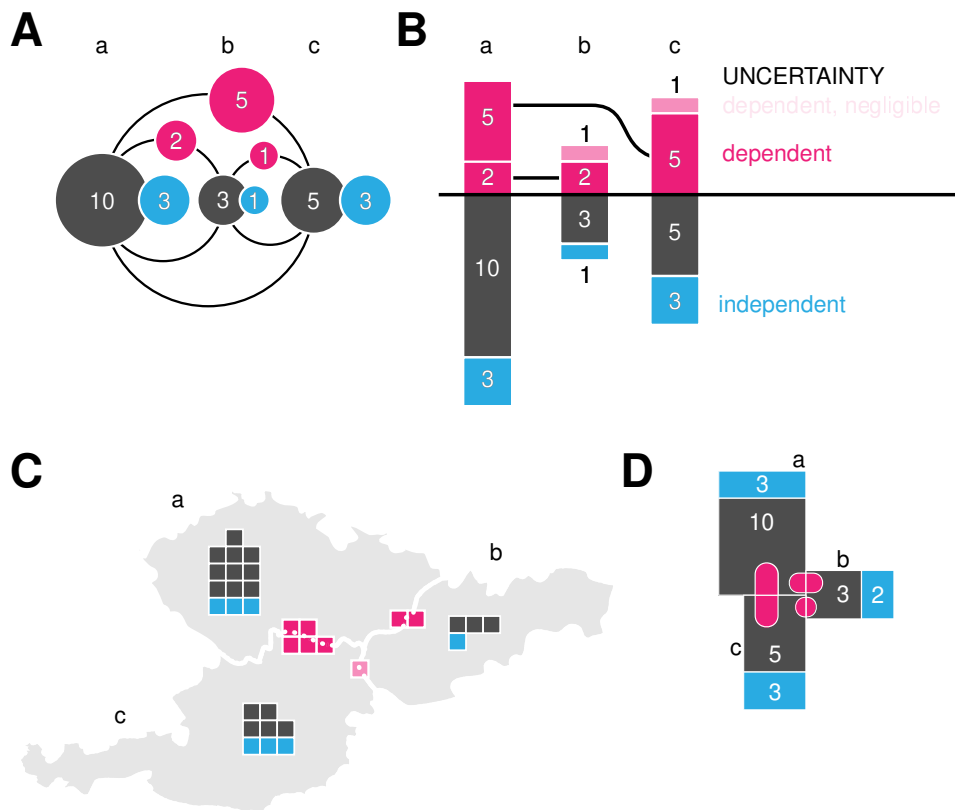
1. For each set, what is its minimum and maximum size, and how much uncertainty is there in the size?
2. What sets can be the largest sets?
3. Between which sets does a large dependent size uncertainty exist?

If the visualization shows the spatial location of the sets as well, we additionally want it to be able to answer the following questions:

1. Where are the smaller or larger regions located?
2. Where are the regions with more uncertainty located?

Discussion

We describe several types of set size uncertainty visualization and some of their affordances and challenges next. To illustrate a basic visualization of such a data set, we use proportional disks drawn in a node-link style (Fig. 2A) where dependencies are indicated with edges/connections between disks.



■ **Figure 11** Four visualizations of set sizes and uncertainties. Certain set sizes are grey, independently uncertain set sizes are blue, and dependently uncertain set sizes are magenta. A. Proportional symbol representation using disks. B. Stacked bar chart representation for the same data as in A. C. Reservoir map visualization with the same underlying (non-spatial) data set as in A. D. Rectangular cartogram.

Topic 1. Sets are commonly represented as disks with proportional size, such as in Euler diagrams. We use a simple visualization to explain the concepts of dependent and independent uncertainty (Fig. 2A). The certain set sizes are represented as gray disks, independent uncertainties as blue disks attached to the gray disks, and dependent uncertainties as pink disks, with lines connecting them to the corresponding gray disks. Though this visualization may introduce some clutter, it allows for showing more dependencies than the other visualizations, by efficiently placing the gray disks in the plane.

Furthermore, it naturally allows for showing dependency between more than two sets, by simply adding extra lines to the additional dependent gray disks.

Topic 2. Bar charts (Fig. 2B) are a simple and effective way to show sizes of sets. It is intuitive to stack uncertain set size bars on top of the bars that show certain sizes. This idea implies that the certain parts show the minimum set size, and we must adopt an additive view of set size. The maximum set size is implied by the length of all bar parts considered together.

Every set has three types of bars that should be distinguishable: a bar for the certain part, a part for the independently uncertain part, and zero or more bars that represent sizes of dependent uncertainties. In case of pairwise uncertainties, these bars always come

in pairs, and we use a line connecting the two bars that are interdependent.

In the visualization, we have the choice of ordering the sets (bars) from left to right in a convenient way. We also have the choice of ordering the dependently uncertain components vertically. These choices influence how complex the connecting lines get, for example, whether they intersect and how long they are.

We think that this type of visualization is the most clear, but it is limited to at most a few dozen sets and it cannot show geographic patterns.

Topic 3. In a second visualization type, we examined rectangular subdivisions like rectangular cartograms and treemaps (Fig. 2D). They allow the visualization to distort the spatial location of the sets and use the available two-dimensional space more efficiently than either bar charts or the reservoir maps when there are many sets. Each set is shown by a rectangle whose size represents the size of the set. If the set represents a geographic region like a country, then these visualizations attempt to maintain adjacencies and relative orientations.

On the border between two rectangles we can show the dependent uncertainties that may exist between adjacent locations. For example, a gas station close to the border between two countries may have clientele from both countries. We may know the total sales of petrol, but not how much exhaust will be caused by this petrol in each of the two countries. Such dependent size uncertainty is shown by a region that overlaps the border and extends on both sides of it.

Independent size uncertainty, which includes uncertainty that is definitely attributable to a specific set (in this case, a location), is shown at the edge of each rectangle, but how a reader will interpret any visualization that does this is unclear. Another issue is that dependent uncertainties between non-adjacent regions are difficult show.

Topic 4. We examine a new style of visualization that we call Reservoir Maps (Fig. 2C). They are suitable when a (geographic) map has regions that represent the sets. They are similar to the visualization using rectangular subdivisions, but here we show regions as they are, and use proportional symbols or enumeration symbols to show the set sizes and their uncertainties. The effect is that we use less space for the visualization of the set size variable, but provide a stronger connection to the actual region (in case it needs to be easily recognizable).

Because the set size corresponding to a region is now symbolized, we can show the certain and independently uncertain parts better inside the region than with rectangular subdivisions. The dependent size uncertainties are again shown in a way that overlaps the border defining the dependency.

The symbol type used for the certain sizes, the independently uncertain sizes, and the dependently uncertain sizes should be the same. We could use a proportional symbol like a disk for each set size component, or an enumeration symbol like small squares. Color can be used to visualize what size is certain and what size is uncertain.

An issue is that dependent uncertainties between non-adjacent regions are difficult show, and it may be challenging to scale symbol sizes for small regions that have large set sizes because this visualization does not distort space.

These visualization types are static. With interaction, many additional options exist for focus and details on demand.

Acknowledgment. The topic was proposed by Wouter Meulemans.

References

- 1 Bilal Alsallakh, Luana Micallef, Wolfgang Aigner, Helwig Hauser, Silvia Miksch, and Peter Rodgers. The state-of-the-art of set visualization. *Computer Graphics Forum*, 35(1):234 – 260, 2016.
- 2 Gerd Gigerenzer. The psychology of good judgment: Frequency formats and simple algorithms. *Medical Decision Making*, 16(3):273–280, 1996. PMID: 8818126.
- 3 Jochen Görtler, Christoph Schulz, Daniel Weiskopf, and Oliver Deussen. Bubble treemaps for uncertainty visualization. *IEEE transactions on visualization and computer graphics*, 24(1):719–728, 2017.
- 4 Jessica Hullman, Xiaoli Qiao, Michael Correll, Alex Kale, and Matthew Kay. In pursuit of error: A survey of uncertainty visualization evaluation. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):903–913, 2019.
- 5 Alexander Lex, Nils Gehlenborg, Hendrik Strobel, Romain Vuillemot, and Hanspeter Pfister. Upset: Visualization of intersecting sets. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1983–1992, 2014.
- 6 Alan M. MacEachren, Robert E. Roth, James O’Brien, Bonan Li, Derek Swingley, and Mark Gahegan. Visual semiotics & uncertainty visualization: An empirical study. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2496–2505, 2012.
- 7 Lace Padilla, Spencer C. Castro, and Helia Hosseinpour. Chapter seven – a review of uncertainty visualization errors: Working memory as an explanatory theory. In Kara D. Federmeier, editor, *The Psychology of Learning and Motivation*, volume 74 of *Psychology of Learning and Motivation*, pages 275–315. Academic Press, 2021.
- 8 Yeseul Park and Jinah Park. Disk diagram: An interactive visualization technique of fuzzy set operations for the analysis of fuzzy data. *Information Visualization*, 9(3):220–232, 2010.
- 9 Max Sondag, Wouter Meulemans, Christoph Schulz, Kevin Verbeek, Daniel Weiskopf, and Bettina Speckmann. Uncertainty treemaps. In *2020 IEEE Pacific Visualization Symposium (PacificVis)*, pages 111–120, 2020.
- 10 Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

Participants

- Daniel Archambault
Swansea University, GB
- Michael Behrisch
Utrecht University, NL
- Susanne Bleisch
FH Nordwestschweiz –
Muttenz, CH
- Annika Bonerath
Universität Bonn, DE
- Steven Chaplick
Maastricht University, NL
- Sara Irina Fabrikant
Universität Zürich, CH
- Amy Griffin
RMIT University –
Melbourne, AU
- Jan-Henrik Haurert
Universität Bonn, DE
- Stephen G. Kobourov
University of Arizona –
Tucson, US
- Martin Krzywinski
BC Cancer Research Centre –
Vancouver, CA
- Eva Mayr
Donau-Universität Krems, AT
- Wouter Meulemans
TU Eindhoven, NL
- Silvia Miksch
TU Wien, AT
- Martin Nöllenburg
TU Wien, AT
- Helen C. Purchase
Monash University –
Clayton, AU
- Peter Rodgers
University of Kent –
Canterbury, GB
- Christian Tominski
Universität Rostock, DE
- Nathan Van Beusekom
TU Eindhoven, NL
- Marc van Kreveld
Utrecht University, NL
- Markus Wallinger
TU Wien, AT
- Bei Wang Phillips
University of Utah –
Salt Lake City, US
- Alexander Wolff
Universität Würzburg, DE
- Hsiang-Yun Wu
FH – St. Pölten, AT



Towards More Flexible and Automated Communication Networks

Artur Hecker^{*1}, Stefan Schmid^{*2}, Henning Schulzrinne^{*3},
Lily Hügerich^{†4}, Sándor Laki^{†5}, and Iosif Salem^{†6}

- 1 Huawei Technologies – München, DE. artur.hecker@huawei.com
- 2 TU Berlin, DE. schmiste@gmail.com
- 3 Columbia University – New York, US. hgs@cs.columbia.edu
- 4 TU Berlin, DE. lily@inet.tu-berlin.de
- 5 Eötvös Lorand University – Budapest, HU. lakis@inf.elte.hu
- 6 TU Berlin, DE. iosif.salem@univie.ac.at

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 22471 “Towards More Flexible and Automated Communication Networks”. Communication networks are becoming more and more automated, allowing to overcome human configuration errors (a frequent reason for outages) and enabling a more fine-grained control, potentially improving also efficiency. For example, the percentage of employees of Telecom companies “really touching the network” is decreasing. The goal of this seminar was to bring together experts in the field to identify and discuss the key challenges in making communication networks more autonomous. To this end, the seminar was structured around a small number of enlightening keynote talks, leaving significant time for breakout sessions and discussions, as well as socializing.

Seminar November 20–23, 2022 – <http://www.dagstuhl.de/22471>

2012 ACM Subject Classification Networks → Network architectures; Networks → Network protocols; Networks → Network algorithms; Software and its engineering

Keywords and phrases networking, communication technologies, automation, programmability, flexibility

Digital Object Identifier 10.4230/DagRep.12.11.96

1 Executive Summary

Artur Hecker

Stefan Schmid

Henning Schulzrinne

License  Creative Commons BY 4.0 International license
© Artur Hecker, Stefan Schmid, and Henning Schulzrinne

Viewed from the perspective of users, communication networks just work fine and have not changed much recently. Under the hood, however, they are currently undergoing substantial changes, which are partly driven by new technologies, such as software-defined networking (SDN) and network function virtualization (NFV), and partly by new requirements, such as reducing operational costs and increasing reliability. SDN and NFV enable programming the behavior of these networks on-demand through software, so that functions and services can be flexibly deployed on short time scales at suitable locations in the network. The drawback of

* Editor / Organizer

† Editorial Assistant / Collector



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

Towards More Flexible and Automated Communication Networks, *Dagstuhl Reports*, Vol. 12, Issue 11, pp. 96–108
Editors: Artur Hecker, Stefan Schmid, Henning Schulzrinne, Lily Hügerich, Sándor Laki, and Iosif Salem



DAGSTUHL
REPORTS

Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

this enhanced flexibility is that, without proper tools, managing and operating such networks becomes more and more challenging. This complexity and the pressures to reduce costs call for largely autonomously operating – or self-driving – networks, i.e. networks with only limited manual intervention.

It is a challenge to provide robust and performant control planes and connectivity in such highly flexible and demanding networking environments, since the notions of control and data plane, by definition, are both related to the notion of the service to be provided. Hence, a network supposed to actively support services with challenging and varying requirements, will also be flexible, and this not just in terms of how it switches flows, but also in terms of the supported queuing models, protocol stacks, deployed and used network and service functions, and even in terms of its own topology. Such a flexible data plane will require an equally flexible control plane capable both of embracing new nodes with new capacities and capabilities and of re-allocating all tasks to new nodes in a shrinking network. The sharing of resources, both in capabilities and in capacities, needs to be efficiently supported not only between different services, but also between the respective control and data planes. Moreover, suitable distributed runtime scheduling algorithms are required in order to utilize and share network resources efficiently and to fulfill highly demanding requirements from certain network slices, e.g., ultra-reliable low latency communication in case of industrial networks. Also, network debugging and diagnostics need to cope with these new demands. It needs to be investigated, to what extent artificial intelligence and machine learning can be applied. This, however, is a rather new topic in the networking domain.

Consequently, the proposed seminar brought together experts from “classical” networking, distributed systems and machine learning for networks.

2 Table of Contents

Executive Summary	
<i>Artur Hecker, Stefan Schmid, and Henning Schulzrinne</i>	96
Challenges	99
Program and Agenda	100
Discussions	100
Day 1 (Lily Huegerich)	100
Day 2 (Iosif Salem)	102
Day 3 (Sándor Laki)	103
Conclusion	104
Overview of Talks	104
Self-Optimizing Network Services	
<i>Gianni Antichi</i>	104
Self-Adjusting Networks	
<i>Chen Avin</i>	105
KIRA – A Scalable ID-based Routing Architecture for Control Planes	
<i>Roland Bless</i>	105
Low-latency Networks Slicing with Programmable Network Elements	
<i>Georg Carle</i>	106
Native Network Intelligence, Fast and Slow	
<i>Dario Rossi</i>	107
Participants	108

3 Challenges

The seminar revolved around the main challenges in network automation and softwarization. There is a wide agreement that communication networks should increasingly operate autonomously without manual intervention. In the context of the Internet and cloud-based systems, first steps in this direction can be observed. Telecommunication infrastructures as still applied in mobile networks slowly move towards virtual network slices and service-based architectures, but do not yet exploit the potential of increased flexibility and elasticity to a full extent. To foster this development, for example, appropriate monitoring and data analysis approaches are needed, as well as methods to describe and handle the “intent”, i.e. to decide, in runtime, how the network should behave in order to fulfill user and service requirements at any given moment in time, topology and protocol.

Generally, a more flexible usage of costly network infrastructures could help to radically cut the service provisioning time while keeping the total cost of ownership low – a very important issue from the network operators’ point of view. This also comprises a flexible usage of network infrastructure across different tenants with data centers of different sizes and at different locations being part of this infrastructure. In some sense, this is comparable to what happened in cloud environments and its data centers during the last years – communication networks still have to emerge along this way. This especially holds for aspects closely related to the network infrastructure itself, e.g., regarding connectivity, scheduling, consistency and the like.

Besides traditional networking techniques, aspects of distributed systems become increasingly important, e.g., with respect to runtime, non-local resource scheduling. Moreover, modelling such complex networks in closed forms (e.g., queueing theory) appears to be increasingly less promising. Focusing, for example, on single TCP connections in the context of congestion control in such emerging complex and more and more automated communication environments appears to have strong limits. Therefore, machine learning attracted more attention in the networking community.

Consequently, autonomously operating and self-driving communication networks can highly profit from an interdisciplinary approach, e.g., including “classical” networking, distributed systems and machine learning.

More specifically, synergies among the following aspects of automation in communication systems are identified:

- Deployment and dynamic adjustment of (virtualized) network and upper-layer services including demand-driven relocation of functionalities and services.
- Robust and performant control planes in highly dynamic and autonomous communication systems that share common networking resource potentially also with the data plane.
- Network debugging and diagnostics (e.g., automated detection of routing failures or DDoS attacks).

These apply to an entire spectrum of networks, including wired/wireless/cellular/hybrid networks. This way, we may be able to discover new synergies and application domains. This follows also the observation that the network landscape is becoming increasingly more diverse considering both, the technological as well as the administrative domain. So this calls in general for concepts that can deal with this increasing diversity.

4 Program and Agenda

To break the ice, we started the seminar with a short round of introductions, where each participant also stated his or her goal for the seminar.

Among other, the following questions came up in this round:

- What are the goals of network automation? Saving human labor? Increasing network efficiency and reliability?
- How automatic and flexible are existing networks?
- Network automation, programmability and SDNs are no longer new – what has worked? What hasn't worked quite as well?
- What are specific missing pieces?
- What are the challenges of doing research in this area?
- What are gaps in practice and understanding?
- In which sense is current practice not “best” practice?
- Where is it still painful, why did certain ideas not work as well as we hoped?
- How should ideas be evaluated? Do they need standards or other infrastructure?

Based on these questions, we structured the breakout sessions according to the time scales, at which network adaptations need to happen:

- N** Network planning and design, spectrum planning [typically weeks]
- C** Configuration, spectrum agility, and programming (data centers, network elements, end systems) [days]
- T** Traffic engineering [hours]
- D** Fault detection and diagnostics [minutes]
- F** Traffic and flow management (congestion, QoS, routing) [seconds and below]

We organized three large breakout sessions, in addition to the discussions during the social program (hike to the castle and excursion):

1. Flexibility = Programmability and observability? What are the required abstractions for automation, what are the programming models (P4 and beyond)? What is the role of network management, what can be useful approaches beyond SNMP? How should configuration languages and approaches evolve?
2. What is the status quo of automation for different network types? Including cellular networks (4G, 5G, 6G, ...), enterprise, home and wired access networks (Ethernet, WLAN, PON, HFC, ...), datacenter networks and cloud (SDN, ...), non-traditional networks (e.g., NGSO satellite, mesh networks, industrial, ...)? What are the pain points and where do we need more research? How can automation benefit security, QoE, cloud configuration, etc.?
3. How can we facilitate research, and how can we educate students in these areas? What is needed to do high-impact research? What infrastructure exists for research and teaching, in terms of testbeds, emulators, simulators? Where are gaps in the curriculum, e.g., on AI/ML, formal verification, (digital) twins, etc.

5 Discussions

5.1 Day 1 (Lily Huegerich)

During the first day, several key themes emerged with regard to the future of network automation. Among these themes were the concepts of intent-driven and self-optimizing networks as well the role of the human in the control loop.

To summarize the many components that fit into these themes we will give a description of a hypothetical system that fulfills the aforementioned qualities. It all starts with the user's intent. The user uses a high-level declarative language of what qualities they want their system to fulfill. Given this and a set of diverse hardware components and constraints, the system is capable of translating the declarative statements of intent into concrete configurations and programs for managing the user's network. The user is able to, but does not need to exercise a high degree of control over the details of the solution, and can specify unchangeable intents as well as less stringent desires for the outcome. The generated configurations and programs are designed to be optimized for the specific hardware on which they will be deployed. The system makes appropriate choices for which functions should run on which network component. These choices take into account the capabilities and limitations of the various components, as well as their relationships to other available components. This helps to ensure that the system is able to make the most effective use of the available hardware.

Here we can see an example of an advanced network automation system that is intent-driven. The user states declaratively and generally what they need and the system from this derives the technicalities. Once the system has a preliminary design the network is able to self-optimize while the network is running. Examples of self-optimizing include adapting switch code to run time traffic or adapting to changing network topologies. The system does this and more in real-time. It is also able to handle network split and rejoining depending on the current state and intent that the network must fulfill. Its self-optimization is in all levels of the network ranging from in-network micro burst management all the way to physical hardware planning. In the case of physical infrastructure, the described system is able to evaluate the feasibility and potential benefits of changes to the user's organization's physical infrastructure. This might include the assessment of the availability of necessary components such as cables, servers, and radio infrastructure, as well as personnel time resources. If a change is deemed beneficial and possible, the system can plan and coordinate every step of the process, including ordering supplies and scheduling technicians.

During this entire process, the human/user is able to remain in the loop. This is because each adaptation in the network is accompanied by an explanation for the change, the reason for it, and the specific modifications made to fulfill the identified requirement. The system as well maintains a clear version control history for the entire system's state, allowing users to revert to a previous state of configuration and programming at any time. The user retains the ability to zoom in and examine the details of the system while it is running, and the system is able to identify specific network components that may be responsible for certain aspects, upon request.

This hypothetical system encompasses many discussed goals, visions, and even existing technologies that were deemed to ease the work of network deployment and management which would bring more flexibility and automation to the field of networking. To bring the state of network management closer to a state were a system like this would be possible, there are a number of areas of research that would be necessary. Including the following:

1. The translation of declarative intent-based statements into concrete configurations and programs. This might be done with natural language processing, or potentially with a declarative programming language.
2. Automation of hardware-specific optimization.
3. Assisting in automizing the currently mainly manual process of human and physical resource management and scheduling needed changes in physical infrastructure.
4. Expand on current network self-optimization research and allow for heterogenous network components to optimize their goals jointly.

5. Find methods that balance and rank the importance of detected anomalies to prevent alarm fatigue while still detecting failures, even grey failures.
6. Expand on network visualization tools.
7. Generate a model that is able to identify the source of a certain network quality. For a simple example, the user would want the reason for packet loss in the network and the system could identify the misconfigured switch responsible.
8. Work on incorporating user-friendly version control in network configurations.

In addition to the technological goals that were discussed during the seminar, the concept of responsible networks was also discussed. This included questions as how to identify or locate the source of certain qualities in the network, such as energy consumption or privacy policy violations. The natural follow to this question is how to blame the network, or how to motivate the controllers of network components to adapt their behavior when these components are deemed responsible for undesirable behavior in the network. This discussion highlights the importance of considering not only the technical aspects of network automation but also the economic and policy factors that can influence the behavior of network actors. The realization of the goals outlined in this text has already begun in various areas. The next steps will involve the expansion of these concepts to multiple areas and for them to cooperate with one another. Lastly, to have all the components smoothly function with one another, have the quality of portability, and a pleasant user interface.

5.2 Day 2 (Iosif Salem)

In the breakout sessions of day 2, we discussed the possible future goals of network automation (NA). We argued that the attempt of categorising NA related problems according to time scale, i.e. problems relevant seconds, minutes, weeks, etc, was not straightforward. However, control loops seem like a good candidate for categorising network management in different time-scales. We then took a step back and discussed what network management (NM) should include. The general context is flexibility and automation. Specifically, NM should include provisioning, performance monitoring/management, fault diagnosis, connection to network (client to cloud/VM, one-to-one/many, bootstrapping), traffic monitoring, security, VNF, CNF, and optimization, to name a few. We also discussed the comparison of NM and SDN (e.g. difference in security requirements). NM can be viewed as a recommendation system, giving alternatives (e.g. via AI) with performance guarantees to the network administrator on how to serve the intended policy in the best way possible. To that end, there could be a hierarchy of management policies. For example, if the network management configuration at level 1 is problematic, the network can always fall back to the level 0 network management configuration (or by keeping a bounded history of snapshots).

We agreed that a main goal of network automation should be to reduce or remove the error prone human work, given that networks are becoming increasingly more complex to manage. Instead, a promising vision would be declarative (or intent-based) network management. This means that the human in the loop should be restricted to declaring the managing policy/requirements, which should be the input to automated NM. That is, humans should be the source and sink of NM, while the remainder of NM should be automated (self-driving networks). In terms of layering, we envisioned a network management layer that focuses on intent. Intent can be expressed in a domain-specific language or using NLP. Types of intent may include: reachability, application performance, security level, encrypted storage, privacy, throughput, responsible networking, etc. The MP also comes with challenges in orchestration

and providing guarantees, or even clashes of (declared) intents or with regulations. Also, a proposed network configuration (computed by the declared intent) might prioritise some customers over others, but should respect SLAs.

5.3 Day 3 (Sándor Laki)

Network intelligence can be split into two main layers: slow and fast thinking. Slow thinking requires pre-existing knowledge, more capacity and computational complexity, and expensive infrastructure. It operates on longer time scales and is responsible for high-level decision-making. In contrast, fast thinking is similar to basic models that can efficiently be executed on constrained hardware but have quick response times. This lower layer often has to make decisions at the packet level on microseconds or shorter time scales. AI-native network automation must combine fast and slow thinking to manage the different control tasks at various levels. Currently, we lack a comprehensive solution, and there are many open questions in this area. The AI models must be interpretable, faithful, and aligned with the regulations (complex law landscape like AI Act in the EU). The maintenance of machine learning models has high algorithmic complexity that is hard to handle with existing tools. Deployment of AI-native solutions is also challenging since heterogeneous hardware is located at different infrastructure locations. The efficient handling of such infrastructure complexity is still an open question, including the optimal placement of AI functions and orchestration. Model execution may have high computational complexity that leads to increased energy consumption. Methods for reducing this complexity and lightweight models with high expressiveness are needed to lower energy consumption, making AI-native approaches greener.

Evaluation of network automation methods requires realistic simulators or testbeds. An ideal testbed should be comprehensive, cover multidomain scenarios, allow testing in the wild with a full stack of various network protocols, enable the emulation of failures, outages, etc., and emulate end-to-end behaviors (even across multiple networking domains). Existing testbeds focus on specific networking aspects, and they could be more user-friendly in many cases. These environments often do not meet the users' expectations. Setting up the environment from scratch is often painful. Tenant resources or the level of access to them is limited. They need to scale better; e.g., they should be able to emulate millions of end users in multiple ISPs/ASes, etc. An ideal platform should also support deep programmability from management to data planes. Automation of the testbed itself is also needed with increased visibility and proof that infrastructure monitoring cannot affect the experiments/measurements. There are good practices in the area of service-meshes that support Canary deployment in production environments. Accordingly, the production environment is not separated from the testbed; e.g., a subset of the traffic (e.g., opt-in users) can be redirected to the new services to validate their correctness. Similar ideas may be applicable in the networking domain, e.g., at academic ISPs. In addition, automated testing of networks can also learn from other areas like chaos engineering. New methods for quantifying network test coverage are also needed for proper evaluation. The smooth deployment of new network releases also poses many challenges, including deployment automation, failure handling, and automated assessment. This may require new network verification methodologies and tools similar to CI-like approaches in traditional software development.

In addition to testbeds, data sets and simulators are also needed for evaluating new methods. Most available traces and data sets in the networking area are very old (e.g., from 1989). Using them in recent papers is not really useful. There is no joint community effort to collect available traces, data sets, and traffic generators at a public repository. If data collection in real environments is not possible, methods need to be developed to create synthetic data, enabling research reproducibility. Obtaining large network topologies is also problematic since, e.g., the largest topology in TopologyZoo only has 794 nodes. To evaluate new network automation methods, we also need to emulate failures that can happen in real-world operations. Failure models are needed for claims of generality. In general, accurate models are needed for topology, traffic, faults, and policies. However, the involvement of consumer ISPs could be required to get accurate data necessary for deriving usable models. Companies need to see the value of exporting their data. Without the involvement of ISPs, we can only see the symptoms without indicating the root cause. For example, RIPE Atlas could see network failures, but the root cause remains hidden. Emulators often introduce side effects in the measurements. These problems must be listed before using a tool or analyzing the results.

6 Conclusion

Overall, the seminar was a great experience and very productive. For many participants it was the first in-person meeting after the pandemic, which made the seminar particularly special and intensive. The many opportunities to meet also in smaller groups sparked many smaller projects, from which we expect several research activities in the next months. Given the success of the seminar and the importance of the topics, we hope we can organize another edition of the seminar in 2-3 years.

7 Overview of Talks

7.1 Self-Optimizing Network Services

Gianni Antichi (Queen Mary University of London, GB)

License  Creative Commons BY 4.0 International license
© Gianni Antichi

Main reference Sebastiano Miano, Alireza Sanaee, Fulvio Risso, Gábor Rétvári, Gianni Antichi: Domain specific run time optimization for software data planes. *ASPLOS 2022*: 1148-1164

URL <https://dl.acm.org/doi/10.1145/3503222.3507769>

State-of-the-art approaches to design, develop and optimize packet-processing programs are based on static compilation: the compiler's input is a description of the forwarding plane semantics and the output is a binary that can accommodate any control plane configuration or input traffic.

In this talk, I will demonstrate that tracking control plane actions and packet-level traffic dynamics at run time opens up new opportunities for code specialization at both software and hardware level. I will introduce a number of new techniques, from static code analysis to adaptive code instrumentation and discuss a toolbox of domain specific optimizations that are not restricted to a specific data plane framework or programming language.

7.2 Self-Adjusting Networks

Chen Avin (Ben Gurion University – Beer Sheva, IL)

License © Creative Commons BY 4.0 International license
© Chen Avin

Joint work of Chen Avin, Stefan Schmid

Main reference Chen Griner, Johannes Zerwas, Andreas Blenk, Manya Ghobadi, Stefan Schmid, Chen Avin:
Cerberus: The Power of Choices in Datacenter Topology Design – A Throughput Perspective. Proc.
ACM Meas. Anal. Comput. Syst. 5(3): 38:1-38:33 (2021)

URL <https://doi.org/10.1145/3491050>

Modern datacenter applications’ bandwidth and latency requirements have led researchers to propose various datacenter topology designs using static, dynamic demand-oblivious (rotor), and/or dynamic demand-aware switches. However, given the diverse nature of datacenter traffic, there is little consensus about how these designs would fare against each other. In this talk, I will present the vision of self-adjusting networks: networks that are optimized toward and “match” the traffic workload they serve. We will discuss information-theoretic metrics to quantify the structure in communication traffic, the achievable performance in datacenter networks matching their demands, present network design principles accordingly, and identify open research challenges.

In addition, I will discuss Cerberus, a unified, two-layer leaf-spine optical datacenter design with three topology types. Cerberus systematically matches different traffic patterns with their most suitable topology type: e.g., latency-sensitive flows are transmitted via a static topology, all-to-all traffic via a rotor topology, and elephant flows via a demand-aware topology. We show analytically and in simulations that Cerberus can improve throughput significantly compared to alternative approaches and operate data centers at higher loads while being throughput-proportional.

7.3 KIRA – A Scalable ID-based Routing Architecture for Control Planes

Roland Bless (KIT – Karlsruher Institut für Technologie, DE)

License © Creative Commons BY 4.0 International license
© Roland Bless

Joint work of Roland Bless, Zoran Despotovic, Artur Hecker, Martina Zitterbart

Main reference R. Bless, M. Zitterbart, Z. Despotovic and A. Hecker, “KIRA: Distributed Scalable ID-based Routing with Fast Forwarding,” 2022 IFIP Networking Conference (IFIP Networking), 2022, doi: 10.23919/IFIPNetworking55013.2022.9829816.

URL <https://doi.org/10.23919/IFIPNetworking55013.2022.9829816>

Future networks will become even more complex than today’s networks. Even very well managed networks recently suffered from large outages due to configuration mistakes made by its operators. In some cases the operators were not able to get back to the routers, because they also lost control to their control plane.

In this talk we present KIRA, a scalable ID-based routing architecture that was specifically designed for control planes. Automation of future networks requires a scalable, zero-touch control plane fabric that interconnects all networked resources (e.g., switches, routers, servers, storage, base stations, ...) providing reliable control access to them. Since future networks are more dynamic due to increased use of virtualization and also usage of mobile nodes like drones, it is useful to have an in-band control solution that does not require a dedicated, separate out-of-band management network, which also requires its own setup, configuration, and management. Furthermore, in-band control allows to let the control plane expand everywhere the network resources come into existence.

KIRA is a scalable routing solution that is tailored to control planes, i.e., in contrast to commonly used routing protocols like OSPF, ISIS, BGP etc., it prioritizes resilient connectivity over route efficiency. It scales to 100,000s of nodes in a single network, it uses ID-based addresses, is zero-touch, and is able to work well in various network topologies. Moreover, it offers a flexible memory/stretch trade-off per node, shows fast recovery from link or node failures, and is loop-free, even during convergence. KIRA is composed of R^2 /Kad in the routing tier and of a PathID-based forwarding scheme in its forwarding tier. R^2 /Kad constructs underlay routes by using a Kademlia-directed ID-based overlay routing strategy and uses source routing between the overlay hops. For control plane packets source routing would induce some per-packet overhead that is avoided by replacing the source routes with PathIDs. PathIDs are similarly used as in label switching and can be partially calculated beforehand. So KIRA establishes a control plane fabric that provides zero-touch connectivity between all networked resources in a scalable manner. Control plane elements such as SDN controllers, Kubernetes controllers, Path Computation Elements, Management servers, and so on can run on top of this control plane fabric. We see such a solution as a foundation that is required for (future) network automation as it constitutes a reliable base for network control that will always work as long as there is physical connectivity available between the entities.

7.4 Low-latency Networks Slicing with Programmable Network Elements

Georg Carle (TU München, DE)

License © Creative Commons BY 4.0 International license
© Georg Carle

Using programmable network components may lead to unwanted performance impairments. This is of particular concern when aiming to provide performance guarantees such as worst-case latencies. In this talk, the issues of how processing architecture, operating system and system configuration, and partitioning of resources among different tenants are addressed. As main method to gain insights on these issues, a data-driven approach with systematic experiments is introduced, which raises the need of reproducibility. As already identified in the workshop on Models, Methods and Tools for Reproducible Network at SIGCOMM 2003, reproducibility is challenging in networked systems research. While significant progress meanwhile has been achieved, further effort remains needed. Flexible traffic generators like MoonGen, and experiment automation with orchestration solutions such as pos, the plain orchestration service, support automated testbed operation, automated experiment execution, and high quality experiment results. Extreme Value Theory is an approach that utilizes real-world measurement data, and that can be used to model predictions on tail latency quantiles on a flow level. This approach is promising for dimensioning networks under latency-constrained service level agreements.

7.5 Native Network Intelligence, Fast and Slow

Dario Rossi (Huawei Technologies – Boulogne-Billancourt, FR)

License © Creative Commons BY 4.0 International license
© Dario Rossi

Joint work of Dario Rossi, Zhang Liang

Main reference Dario Rossi, Liang Zhang: “Network artificial intelligence, fast and slow”, in Proc. of the 1st International Workshop on Native Network Intelligence, NativeNi 2022, Rome, Italy, 9 December 2022, pp. 14–20, ACM, 2022.

URL <https://doi.org/10.1145/3565009.3569521>

Main reference Dario Rossi, Liang Zhang: “Landing AI on Networks: An Equipment Vendor Viewpoint on Autonomous Driving Networks”, IEEE Trans. Netw. Serv. Manag., Vol. 19(3), pp. 3670–3684, 2022.

URL <https://doi.org/10.1109/TNSM.2022.3169988>

As networks have historically been built around connectivity, architectural features concerning quality of service, mobility, security and privacy have been added as afterthoughts – with consequent well known architectural headaches for their later integration. Despite Artificial Intelligence (AI) is more a means to an end, that an architectural feature itself, this is not completely different from what concerns its integration: in particular, while Cloud and Edge computing paradigms made it possible to use AI techniques to relieve part of network operation, however AI is currently little more than an additional tool. This talk describes a vision of future programmable networks, where AI becomes a first class commodity: its founding principle lays around the concept of “fast and slow” [1] types of AI reasoning, each of which offers different types of AI capabilities to process network data – in particular to make use of existing knowledge, or to create and extend the knowledge itself. Introducing emerging AI-to-AI communication patterns as we move towards more intelligent networks, we also outline desirable properties (explainable automated, fit and sustainable) of AI native networks.

References

- 1 D. Kanhehan, “Thinking, fast and slow”, 2011

Participants

- Gianni Antichi
Queen Mary University of
London, GB
- Chen Avin
Ben Gurion University –
Beer Sheva, IL
- Roland Bless
KIT – Karlsruher Institut für
Technologie, DE
- Georg Carle
TU München, DE
- Klaus-Tycho Foerster
TU Dortmund, DE
- Fabien Geyer
Airbus – München, DE
- Sergey Gorinsky
IMDEA Networks Institute –
Madrid, ES
- David Hay
The Hebrew University of
Jerusalem, IL
- Artur Hecker
Huawei Technologies –
München, DE
- Lily Hügerich
TU Berlin, DE
- Karin Anna Hummel
Johannes Kepler Universität
Linz, AT
- Sándor Laki
Eötvös Lorand University –
Budapest, HU
- Shir Landau Feibish
The Open University of Israel –
Raanana, IL
- Huiran Liu
TU Berlin, DE
- Gábor Rétvári
Budapest University of
Technology & Economics, HU
- Dario Rossi
Huawei Technologies –
Boulogne-Billancourt, FR
- Iosif Salem
TU Berlin, DE
- Gabriel Scalosub
Ben Gurion University –
Beer Sheva, IL
- Stefan Schmid
TU Berlin, DE
- Henning Schulzrinne
Columbia University –
New York, US
- Cigdem Sengul
Brunel University – London, GB
- Martina Zitterbart
KIT – Karlsruher Institut für
Technologie, DE



Vertex Partitioning in Graphs: From Structure to Algorithms

Maria Chudnovsky^{*1}, Neeldhara Misra^{*2}, Daniel Paulusma^{*3},
Oliver Schaudt^{*4}, and Akanksha Agrawal^{†5}

1 Princeton University, US. mchudnov@math.princeton.edu

2 IIT Gandhinagar, India. neeldhara.m@iitgn.ac.in

3 Durham University, GB. daniel.paulusma@durham.ac.uk

4 Bayer AG – Leverkusen, DE. oliver.schaudt@bayer.com

5 Indian Institute of Technology Madras, IN. akanksha.agrawal.2029@gmail.com

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 22481 “Vertex Partitioning in Graphs: From Structure to Algorithms”, which was held from 27 November to 2 December 2023. The report contains abstracts for presentations about recent structural and algorithmic developments for a variety of vertex partitioning problems. It also contains a collection of open problems which were posed during the seminar.

Seminar November 27 – December 2, 2022 – <http://www.dagstuhl.de/22481>

2012 ACM Subject Classification Mathematics of computing → Graph algorithms; Theory of computation → Problems, reductions and completeness

Keywords and phrases computational complexity, hereditary graph classes, parameterized algorithms, polynomial-time algorithms, vertex partitioning

Digital Object Identifier 10.4230/DagRep.12.11.109

1 Executive Summary

Maria Chudnovsky (Princeton University, US)

Neeldhara Misra (IIT Gandhinagar, India)

Daniel Paulusma (Durham University, GB)

Oliver Schaudt (Bayer AG – Leverkusen, DE)

License © Creative Commons BY 4.0 International license

© Maria Chudnovsky, Neeldhara Misra, Daniel Paulusma, and Oliver Schaudt

Many important discrete optimization problems can be modelled as graph problems that ask if the set of vertices in a graph can be partitioned into a smallest number of sets, such that each set has the same property, or into some number of sets, such that each set has a specific property of their own. This leads to a rich framework of vertex partitioning problems, which include classical problems such as GRAPH COLOURING, GRAPH HOMOMORPHISM, VERTEX COVER, FEEDBACK VERTEX SET and ODD CYCLE TRANSVERSAL, and variants and generalizations of these problems.

Most vertex partitioning problems are computationally hard. The central research aim of our seminar was to increase our understanding of the computational complexity of these problems. The main approach followed at the seminar for achieving this was to restrict the input of some problem to some special graph class. The fundamental question then becomes

* Editor / Organizer

† Editorial Assistant / Collector



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

Vertex Partitioning in Graphs: From Structure to Algorithms, *Dagstuhl Reports*, Vol. 12, Issue 11, pp. 109–123

Editors: Maria Chudnovsky, Misra, Neeldhara, Daniel Paulusma, Oliver Schaudt, and Akanksha Agrawal



DAGSTUHL
REPORTS

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

whether such a restriction makes the problem tractable or whether the problem remains hard. In order to approach this question, we followed a systematic way by considering graph classes characterized by some finite family \mathcal{H} of obstructions (as induced subgraph, subgraph, minor etc.).

In line with the seminar's research aim, the seminar brought together researchers from Discrete Mathematics, working in structural graph theory, and researchers from Theoretical Computer Science, working in algorithmic graph theory. In total, 36 participants from 12 different countries attended the seminar.

The scientific program of the seminar consisted of 23 sessions: 5 survey talks of fifty minutes, 13 contributed talks of at most thirty minutes and 5 open problem sessions. This left ample time for discussions and problem solving. Participants presented the progress that made during the workshop during several "progress report" sessions. One of the questions discussed at the workshop was a long-time open problem concerning the complexity of detecting whether a graph contains two long induced cycles with no edges between them. In the course of the workshop Khang Le informed us that he found a polynomial-time algorithm to solve this problem; Le's proof was presented by Paul Seymour.

Each of the five survey talks covered a particular structural or algorithmic key aspect of the seminar in order to enable collaborations of researchers with different backgrounds. On Monday, Paul Seymour presented a number of recent developments on the Erdős-Hajnal conjecture including several open problems. On the same day, Tara Abrishami described a variety of techniques for proving the boundedness or unboundedness of treewidth of hereditary graph classes. On Tuesday, Daniel Lokshtanov explained how algorithms for the INDEPENDENT SET problem on P_k -free graphs developed over time, and also gave extensions of these results to other graph classes. On the same day, Daniel Král' surveyed basic results and open problems for two classical graph colouring parameters, the fractional and circular chromatic number of a graph, and one recent graph colouring parameter, the gyrochromatic number. On Wednesday, Henning Bruhn-Fujimoto gave a survey talk on Erdős-Pósa type questions, which relate to graph packing and graph covering dualities. In this talk, many open problems were given as well.

The five general open problem sessions took place on Monday, Tuesday and Wednesday. Details of the presented problems can be found in the report, together with abstracts of all the talks.

We are grateful to Gerhard Woeginger for all his help with our seminar when it was originally planned to take place from 31 January to 5 February 2021. Our seminar was postponed to November 2022 due to the pandemic, and sadly, Gerhard passed away on 1 April 2022.

We also thank Akanksha Agrawal for her help with the Dagstuhl report of our seminar.

2 Table of Contents

Executive Summary

Maria Chudnovsky, Neeldhara Misra, Daniel Paulusma, and Oliver Schaudt 109

Overview of Talks

Induced subgraphs and tree decompositions <i>Tara Abrishami</i>	113
Minimum k -permutation-avoiding words <i>Bogdan Alecu</i>	113
Erdős-Pósa type questions <i>Henning Bruhn-Fujimoto</i>	113
Characterizing Graphs with Few Minimal Separators <i>Peter Gartland</i>	114
Vertex Deletion Parameterized by Elimination Distance and Even Less <i>Bart Jansen</i>	114
Computing Tree Decompositions with Small Independence Number <i>Tuukka Korhonen</i>	115
Rational relaxations of chromatic number <i>Daniel Král'</i>	115
Independent Set on P_k -free graphs and beyond. <i>Daniel Lokshantov</i>	116
Complexity Framework for Forbidden Subgraphs: When Hardness Is Not Preserved under Edge Subdivision <i>Barnaby Martin</i>	116
Proving a directed analogue of the Gyárfás-Sumner conjecture for orientations of P_4 <i>Tomás Masarík</i>	117
Parameterized Complexity of Streaming Diameter <i>Jelle Oostveen</i>	117
Complexity of problems efficiently solvable on subcubic graphs <i>Sukanya Pandey</i>	118
Computing maximum weight independent set via dynamic programming: How much we can relax the notion of a potential maximal clique? <i>Marcin Pilipczuk</i>	118
Cograph colorings of path- and antipath-free graphs, with applications <i>Michał Pilipczuk</i>	119
Understanding graphs with no long claws <i>Paweł Rzążewski</i>	119
A survey of recent developments on the Erdős-Hajnal conjecture <i>Paul Seymour</i>	119
Complexity Framework for Forbidden Subgraphs <i>Siani Smith</i>	120

Steiner Forest on H -subgraph-free graphs <i>Erik Jan van Leeuwen</i>	120
Open problems	
Logarithmic degree and logarithmic treewidth <i>Maria Chudnovsky</i>	121
Parameterized complexity of k -Coloring on P_5 -free graphs <i>Petr A. Golovach</i>	121
Finding H -Induced Minors <i>Daniel Paulusma</i>	122
k -Community <i>Jan Arne Telle</i>	122
Participants	123

3 Overview of Talks

3.1 Induced subgraphs and tree decompositions

Tara Abrishami (Princeton University, US)

License  Creative Commons BY 4.0 International license
© Tara Abrishami

A classic result from Robertson and Seymour’s Graph Minors Project is the Grid Minor Theorem, which states that a graph has large treewidth if and only if it has a large grid minor. The Grid Minor Theorem characterizes the relationship between bounded treewidth and minors. Recently, much research on treewidth has focused on the relationship between bounded treewidth and induced subgraphs, aiming for an analog of the Grid Minor Theorem for induced subgraphs. Currently, there are several results that identify hereditary graph classes of bounded treewidth, but no complete characterization has yet been found. In this talk, we review recent progress and results involving induced subgraph obstructions to bounded treewidth.

3.2 Minimum k -permutation-avoiding words

Bogdan Alecu (University of Leeds, GB)

License  Creative Commons BY 4.0 International license
© Bogdan Alecu

Joint work of Bogdan Alecu, Tara Abrishami, Maria Chudnovsky, Sepehr Hajebi, Sophie Sprinkl


Call a graph G *linear* if $V(G)$ admits a partition into two sets S and P such that:

- S is stable;
- P induces a path;
- every vertex of P has at most one neighbour in S .

Inspired by the study of treewidth (in the hereditary closure) of linear graphs, we formulate a word-combinatorial problem concerning the minimum length of words with certain properties. We describe the beginning of a solution to the problem.

3.3 Erdős-Pósa type questions

Henning Bruhn-Fujimoto (Universität Ulm, DE)

License  Creative Commons BY 4.0 International license
© Henning Bruhn-Fujimoto

Joint work of Henning Bruhn-Fujimoto, Matthias Heinlein, Felix Joo, Arthur Ulmer

Main reference Henning Bruhn, Matthias Heinlein, Felix Joo: “The Edge-Erdős-Pósa Property”, *Comb.*, Vol. 41(2), pp. 147–173, 2021.



URL <https://doi.org/10.1007/s00493-020-4071-7>

There are many examples of a packing and covering duality: either there are many disjoint A – B -paths (a packing), or there is a small vertex set meeting all A – B -paths (a covering); either there are many disjoint cycles or a small cycle cover; either there are many disjoint K_4 -subdivisions or a small vertex set meeting them all, and so on.

In this survey talk I will discuss several such packing and covering dualities. We will look at variants, for example, when the graphs is additionally endowed with group edge weights, and I will try to present as many open problems as possible.

3.4 Characterizing Graphs with Few Minimal Separators

Peter Gartland (University of California – Santa Barbara, US)

License  Creative Commons BY 4.0 International license
 Peter Gartland

Joint work of Peter Gartland, Daniel Lokshtanov

A class \mathcal{F} of graphs is called tame if every graph in \mathcal{F} on n vertices contains at most $n^{O(1)}$ minimal separators, *quasi-tame* if every graph in \mathcal{F} on n vertices contains at most $2^{\log^{O(1)}(n)}$ minimal separators, and *feral* if there exists a constant $c > 1$ so that \mathcal{F} contains n -vertex graphs with at least c^n minimal separators for arbitrarily large n . The classification of graph classes into (quasi-) tame or feral has numerous algorithmic consequences, and has recently received considerable attention.

In this talk we precisely characterize the structure of graphs which have few minimal separators. Specifically we show that every graph which excludes certain graphs called *k-creatures* and *k-critters* as induced subgraphs has at most quasi-polynomially many minimal separators. We then demonstrate that this sufficient condition for having few minimal separators is the “right” one. In particular we show that every hereditary graph class \mathcal{F} definable in CMSO logic that contains *k-creatures* or *k-critters* for every k is feral.

3.5 Vertex Deletion Parameterized by Elimination Distance and Even Less

Bart Jansen (TU Eindhoven, NL)

License  Creative Commons BY 4.0 International license
 Bart Jansen

Joint work of Bart Jansen, Jari J.H. de Kroon, Michal Włodarczyk

Main reference Bart M. P. Jansen, Jari J. H. de Kroon, Michal Włodarczyk: “Vertex deletion parameterized by elimination distance and even less”, in Proc. of the STOC ’21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021, pp. 1757–1769, ACM, 2021.

URL <https://doi.org/10.1145/3406325.3451068>

We study the parameterized complexity of various classic vertex-deletion problems such as Odd cycle transversal, Vertex planarization, and Chordal vertex deletion under hybrid parameterizations. Existing FPT algorithms for these problems either focus on the parameterization by solution size, detecting solutions of size k in time $f(k) \cdot n^{O(1)}$, or width parameterizations, finding arbitrarily large optimal solutions in time $f(w) \cdot n^{O(1)}$ for some width measure like treewidth. We unify these lines of research by presenting FPT algorithms for parameterizations that can simultaneously be arbitrarily much smaller than the solution size and the treewidth.

The first class of parameterizations is based on the notion of elimination distance of the input graph to the target graph class H , which intuitively measures the number of rounds needed to obtain a graph in H by removing one vertex from each connected component in each round. The second class of parameterizations consists of a relaxation of the notion

of treewidth, allowing arbitrarily large bags that induce subgraphs belonging to the target class of the deletion problem as long as these subgraphs have small neighborhoods. Both kinds of parameterizations have been introduced recently and have already spawned several independent results.

Our contribution is twofold. First, we present a framework for computing approximately optimal decompositions related to these graph measures. Namely, if the cost of an optimal decomposition is k , we show how to find a decomposition of cost $k^{O(1)}$ in time $f(k) \cdot n^{O(1)}$. This is applicable to any class \mathcal{H} for which we can solve the so-called separation problem. Secondly, we exploit the constructed decompositions for solving vertex-deletion problems by extending ideas from algorithms using iterative compression and the finite state property. For the three mentioned vertex-deletion problems, and all problems which can be formulated as hitting a finite set of connected forbidden (a) minors or (b) (induced) subgraphs, we obtain FPT algorithms with respect to both studied parameterizations. For example, we present an algorithm running in time $n^{O(1)} + 2^{(k^{O(1)}) \cdot (n+m)}$ and polynomial space for Odd cycle transversal parameterized by the elimination distance k to the class of bipartite graphs.

3.6 Computing Tree Decompositions with Small Independence Number

Tuukka Korhonen (University of Bergen, NO)

License © Creative Commons BY 4.0 International license
© Tuukka Korhonen

Joint work of Clément Dallard, Fedor V. Fomin, Petr A. Golovach, Tuukka Korhonen, Martin Milanič

Main reference Clément Dallard, Fedor V. Fomin, Petr A. Golovach, Tuukka Korhonen, Martin Milanič: “Computing Tree Decompositions with Small Independence Number”, CoRR, Vol. abs/2207.09993, 2022.

URL <https://doi.org/10.48550/arXiv.2207.09993>

The independence number of a tree decomposition is the maximum of the independence numbers of the subgraphs induced by its bags. The tree-independence number of a graph is the minimum independence number of a tree decomposition of it. Several NP-hard graph problems, like maximum weight independent set, can be solved in polynomial time if the input graph is given with a tree decomposition of bounded independence number. I discuss results about computing the tree-independence number of a graph, and also its relation to a graph parameter “minor-matching hypertree width” of Yolov [SODA 2018].

3.7 Rational relaxations of chromatic number

Daniel Král’ (Masaryk University – Brno, CZ)

License © Creative Commons BY 4.0 International license
© Daniel Král’

Joint work of Pablo Candela, Carlos Catalá, Robert Hancock, Adam Kabela, Daniel Král’, Ander Lamaison, Lluís Vena

Main reference Pablo Candela, Carlos Catalá, Robert Hancock, Adam Kabela, Daniel Král’, Ander Lamaison, Lluís Vena: “Coloring graphs by translates in the circle”, Eur. J. Comb., Vol. 96, p. 103346, 2021.


URL <https://doi.org/10.1016/j.ejc.2021.103346>

The fractional and circular chromatic numbers are the two most well-known rational relaxations of the chromatic number of a graph. During the talk, we introduce the notion of gyrocoloring, a new recent relaxation of the chromatic number. Gyrocoloring of graphs stems from the notion of a coloring base studied by Avila and Candela in ergodic theory. We present various results concerning gyrocolorings of graphs, in particular, we show that

the gyrochromatic chromatic is sandwiched between the fractional and circular chromatic numbers and it is robust in the sense that it permits various alternative definitions, analogous to alternative definitions of fractional and circular chromatic numbers.

3.8 Independent Set on P_k -free graphs and beyond.

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

License  Creative Commons BY 4.0 International license
© Daniel Lokshantov

Joint work of Peter Gartland, Daniel Lokshantov, Marcin Pilipczuk, Michal Pilipczuk, Pavel Rzázewski

Main reference Peter Gartland, Daniel Lokshantov: “Independent Set on C_{\geq} -Free Graphs in Quasi-Polynomial Time”, CoRR, Vol. abs/2007.11402, 2020.

URL <https://arxiv.org/abs/2007.11402>

In this talk I will survey fairly recent (past 10 years or so) developments for algorithms for Independent Set on P_k -free graphs, as well as extensions to other graph classes (excluding long cycles or long claws as induced subgraphs), and to other problems (such as Feedback Vertex Set or 3-Coloring).

3.9 Complexity Framework for Forbidden Subgraphs: When Hardness Is Not Preserved under Edge Subdivision

Barnaby Martin (Durham University, GB)

License  Creative Commons BY 4.0 International license
© Barnaby Martin

Joint work of Sukanya Pandey, Daniel Paulusma, Siani Smith, Erik Han van Leeuwen

Main reference Barnaby Martin, Sukanya Pandey, Daniël Paulusma, Siani Smith, Erik Jan van Leeuwen:

“Complexity Framework for Forbidden Subgraphs: When Hardness Is Not Preserved under Edge Subdivision”, CoRR, Vol. abs/2211.14214, 2022.

URL <https://doi.org/10.48550/arXiv.2211.14214>

A graph G is H -subgraph-free if G does not contain H as a (not necessarily induced) subgraph. We make inroads into the classification of three problems for H -subgraph-free graphs that have the properties that they are solvable in polynomial time on classes of bounded treewidth and NP-complete on subcubic graphs, yet NP-hardness is not preserved under edge subdivision. The three problems are k -Induced Disjoint Paths, C_5 -Colouring and Hamilton Cycle. Although we do not complete the classifications, we show that the boundary between polynomial time and NP-complete differs for C_5 -Colouring from the other two problems.

3.10 Proving a directed analogue of the Gyárfás-Sumner conjecture for orientations of P_4

Tomás Masarík (University of Warsaw, PL)

License © Creative Commons BY 4.0 International license
© Tomás Masarík

Joint work of Linda Cook, Tomás Masarík, Marcin Pilipczuk, Amadeus Reinald, Uéverton S. Souza

Main reference Linda Cook, Tomás Masarík, Marcin Pilipczuk, Amadeus Reinald, Uéverton S. Souza: “Proving a directed analogue of the Gyárfás-Sumner conjecture for orientations of P_4 ”, CoRR, Vol. abs/2209.06171, 2022.

URL <https://doi.org/10.48550/arXiv.2209.06171>

An oriented graph is a digraph that does not contain a directed cycle of length two. An (oriented) graph D is H -free if D does not contain H as an induced sub(di)graph. The Gyárfás-Sumner conjecture is a widely-open conjecture on simple graphs, which states that for any forest F , there is some function f such that every F -free graph G with clique number $\omega(G)$ has chromatic number at most $f(\omega(G))$. Aboulker, Charbit, and Naserasr [Extension of Gyárfás-Sumner Conjecture to Digraphs; E-JC 2021] proposed an analogue of this conjecture to the dichromatic number of oriented graphs. The dichromatic number of a digraph D is the minimum number of colors required to color the vertex set of D so that no directed cycle in D is monochromatic.

Aboulker, Charbit, and Naserasr’s $\vec{\chi}$ -boundedness conjecture states that for every oriented forest F , there is some function f such that every F -free oriented graph D has dichromatic number at most $f(\omega(D))$, where $\omega(D)$ is the size of a maximum clique in the graph underlying D . In this paper, we perform the first step towards proving Aboulker, Charbit, and Naserasr’s $\vec{\chi}$ -boundedness conjecture by showing that it holds when F is any orientation of a path on four vertices.

3.11 Parameterized Complexity of Streaming Diameter

Jelle Oostveen (Utrecht University, NL)

License © Creative Commons BY 4.0 International license
© Jelle Oostveen

Joint work of Jelle J. Oostveen, Erik Jan van Leeuwen

Main reference Jelle J. Oostveen, Erik Jan van Leeuwen: “Parameterized Complexity of Streaming Diameter and Connectivity Problems”, in Proc. of the 17th International Symposium on Parameterized and Exact Computation, IPEC 2022, September 7-9, 2022, Potsdam, Germany, LIPIcs, Vol. 249, pp. 24:1–24:16, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

URL <https://doi.org/10.4230/LIPIcs.IPEC.2022.24>

In this talk, we consider the parameterized complexity of Diameter in the streaming paradigm. The streaming paradigm is a model where our graph is not in memory, but we inspect it through the so-called “stream”. A focus on memory-efficiency is crucial in this setting. On the positive end, knowing a vertex cover of size k allows for algorithms in the Adjacency List (AL) streaming model whose number of passes is constant and memory is $O(\log n)$ for any fixed k . On the negative end, many other parameters lead to lower bounds in the AL model, where $\Omega(n/p)$ bits of memory is needed for any p -pass algorithm even for constant parameter values. In particular, this holds for graphs with a known modulator (deletion set) of constant size to a graph that has no induced subgraph isomorphic to a fixed graph H , for most H . Open problems for our work include solving Diameter for specific H -free graphs, and for interval graphs in the AL model. Another interesting open problem relevant to this work is whether there is a streaming algorithm for Vertex Cover $[k]$ using $O(\text{poly}(k))$ passes and $O(\text{poly}(k, \log n))$ bits of memory.

3.12 Complexity of problems efficiently solvable on subcubic graphs

Sukanya Pandey (Utrecht University, NL)

License  Creative Commons BY 4.0 International license
© Sukanya Pandey

Joint work of Matthew Johnson, Barnaby Martin, Sukanya Pandey, Daniël Paulusma, Siani Smith, Erik Jan van Leeuwen

We proposed a framework for the complexity classification of problems that satisfy the following three conditions:

1. they are efficiently computable on graph classes of bounded treewidth
2. they are computationally hard on the class of subcubic graphs
3. they remain computationally hard on subdivisions of subcubic graphs

We call the problems that satisfy all three conditions C123-problems. This talk discusses the complexity of problems that satisfy the first condition but violate the second on graph classes excluding certain subdivided stars as subgraphs. A few notable examples of these problems are feedback vertex set, independent feedback vertex set, and colouring. In particular, we show that certain classes of connected, H -subgraph-free graphs that are not subcubic must have bounded treewidth. We show this for the cases when $H = S_{1,1,1,r}$ and $H = S_{1,1,q,r}$. The immediate consequence of these theorems is that the problems under consideration are polynomial-time solvable on the aforementioned graph classes. We also discuss the NP-completeness of Feedback Vertex Set and Independent Feedback Vertex Set on $S_{2,2,2,2}$ -subgraph-free graphs, leaving $S_{1,p,q,r}$ -subgraph-free graphs as the only open case remaining.

3.13 Computing maximum weight independent set via dynamic programming: How much we can relax the notion of a potential maximal clique?

Marcin Pilipczuk (University of Warsaw, PL)

License  Creative Commons BY 4.0 International license
© Marcin Pilipczuk

Joint work of Maria Chudnovsky, Rose McCarty, Marcin Pilipczuk, Michał Pilipczuk, Paweł Rzażewski

In this tutorial talk I presented the framework of potential maximal cliques, with emphasis on recent attempts to relax the requirements for the enumerated family of “approximate” potential maximal cliques to containers and carvers. This enables solving a large family of problems in six-vertex-path-free graphs in polynomial time

3.14 Cograph colorings of path- and antipath-free graphs, with applications

Michał Pilipczuk (University of Warsaw, PL)

License © Creative Commons BY 4.0 International license
© Michał Pilipczuk

Joint work of Patrice Ossona de Mendez, Michał Pilipczuk, Sebastian Siebertz

Main reference Patrice Ossona de Mendez, Michał Pilipczuk, Sebastian Siebertz: “Transducing paths in graph classes with unbounded shrubdepth”, *European Journal of Combinatorics*, p. 103660, 2022.

URL <https://doi.org/10.1016/j.ejc.2022.103660>

We prove that if a graph G excludes a path, complement of a path, and a threshold graph, all as induced subgraphs, then G can be colored with a bounded number of colors so that every color induces a cograph. This statement, together with its depth-2 generalization that concerns pairs of colors, can be used to prove a conjecture about the characterization of graph classes of bounded shrubdepth through forbidden First-Order transductions. At the end we will discuss variants of this conjecture for linear cliquewidth and cliquewidth, which remain open.

3.15 Understanding graphs with no long claws

Paweł Rzażewski (Warsaw University of Technology, PL)

License © Creative Commons BY 4.0 International license
© Paweł Rzażewski

Joint work of Tara Abrishami, Maria Chudnovsky, Cemil Dibek, Konrad Majewski, Tomáš Masarík, Jana Novotná, Karolina Okrasa, Marcin Pilipczuk, Paweł Rzażewski, Marek Sokolowski

Main reference Konrad Majewski, Tomáš Masarík, Jana Novotná, Karolina Okrasa, Marcin Pilipczuk, Paweł Rzażewski, Marek Sokolowski: “Max Weight Independent Set in Graphs with No Long Claws: An Analog of the Gyárfás’ Path Argument”, in *Proc. of the 49th International Colloquium on Automata, Languages, and Programming, ICALP 2022, July 4-8, 2022, Paris, France, LIPIcs*, Vol. 229, pp. 93:1–93:19, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

URL <https://doi.org/10.4230/LIPIcs.ICALP.2022.93>

A classic result of Alekseev asserts that for connected H the Max Independent Set (MIS) problem in H -free graphs is NP-hard unless H is a path or a subdivided claw. Recently we have witnessed some great progress in understanding the complexity of MIS in P_t -free graphs. The situation for forbidden subdivided claws is, however, much less understood.

During the talk we will present some recent advances in understanding the structure of graphs with no long induced claws, and their applications in designing algorithms for MIS and related problems.

3.16 A survey of recent developments on the Erdős-Hajnal conjecture

Paul Seymour (Princeton University, US)

License © Creative Commons BY 4.0 International license
© Paul Seymour

A graph G is “ H -free” if no induced subgraph is isomorphic to H . In 1977, Erdős and Hajnal made the conjecture that, for every graph H , there exists $c > 0$ such that every H -free graph G has a clique or stable set of size at least $|G|^c$; and they proved that this is true with $|G|^c$ replaced by $2^{c\sqrt{\log |G|}}$. Until now, there has no improvement on this result (for general H).

In joint work with M. Bucić, T. Nguyen, A. Scott, we recently proved a strengthening: that for every graph H , there exists $c > 0$ such that every H -free graph G with $|G| \geq 2$ has a clique or stable set of size at least

$$2^{c\sqrt{\log |G| \log \log |G|}}.$$

Indeed, we prove the corresponding strengthening of a theorem of Fox and Sudakov, which in turn was a common strengthening of theorems of Rödl, Nikiforov, and the theorem of Erdős and Hajnal mentioned above.

In this talk, we survey this material, and a number of other results related to the Erdős-Hajnal conjecture, for instance:

- it holds when H is the five-vertex cycle (a recent result joint with M. Chudnovsky, A. Scott and S. Spirkl);
- it “almost” holds when H is the five-vertex path P_5 (Blanco and Bucić recently proved that every P_5 -free graph G has a clique or stable set of size at least $2^{c(\log |G|)^{2/3}}$);
- it almost holds in another sense when $H = P_5$; Chudnovsky, Scott and the speaker proved that every P_5 -free graph G has chromatic number at most $k^{\log k}$, where k is the clique number of G .

3.17 Complexity Framework for Forbidden Subgraphs

Siani Smith (University of Bristol, GB)

License © Creative Commons BY 4.0 International license
© Siani Smith

Joint work of 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>

We present a framework to classify the complexity of certain problems on H -subgraph-free graph classes. A graph problem Π is a C123-problem if it satisfies the following three conditions:

- C1.** Π is efficiently solvable for every graph class of bounded treewidth;
- C2.** Π is computationally hard for the class of subcubic graphs; and
- C3.** Π is computationally hard under edge subdivision of subcubic graphs.

In this talk we give a meta-theorem classifying the complexity of problems satisfying these conditions. We then discuss the limits of this framework together with a number of open problems.

3.18 Steiner Forest on H -subgraph-free graphs

Erik Jan van Leeuwen (Utrecht University, NL)

License © Creative Commons BY 4.0 International license
© Erik Jan van Leeuwen

Joint work of Hans L. Bodlaender, Matthew Johnson, Barnaby Martin, Jelle J. Oostveen, Sukanya Pandey, Daniel Paulusma, Siani Smith, Erik Jan Leeuwen

In this talk, we consider the complexity of STEINER FOREST on graphs that exclude a fixed graph H as a subgraph. Recently, a framework was proposed to classify the complexity of

problems on H -subgraph-free graphs (see [1] as well as the talk in this seminar). Unfortunately, STEINER FOREST does not fit this framework. In this talk, we discuss why this is the case as well as our progress on obtaining a separate classification for STEINER FOREST.


References

- 1 Matthew Johnson, Barnaby Martin, Jelle J. Oostveen, Sukanya Pandey, Daniël Paulusma, Siani Smith, and Erik Jan van Leeuwen. *Complexity Framework For Forbidden Subgraphs*. arXiv:2211.12887 [math.CO], 2022.

4 Open problems

4.1 Logarithmic degree and logarithmic treewidth

Maria Chudnovsky (Princeton University, US)

License  Creative Commons BY 4.0 International license
© Maria Chudnovsky

Joint work of Tara Abrishami, Bogdan Alecu, Sepehr Hajebi, Sophie Spirkl

Let t be an integer. A graph is *clean* if it does not contain any of the following as an induced subgraph.

- the complete graph on t vertices
- the complete bipartite graph $K_{t,t}$
- a subdivision of the $t \times t$ -wall
- the line graph of a subdivision of a $t \times t$ wall.

Is the following true: for every integer t there exists an integer $c(t)$ such that every t -clean n -vertex graph with maximum degree at most $\log n$ has treewidth at most $c(t) \log n$?

4.2 Parameterized complexity of k -Coloring on P_5 -free graphs

Petr A. Golovach (University of Bergen, NO)

License  Creative Commons BY 4.0 International license
© Petr A. Golovach


Only very few parameterized complexity results for COLORING on H -free graphs are known (see [1] for the survey) and some of the open problems seem to be highly nontrivial. In particular, the following problem, which was first stated by Hoáng et al. in [2], remains open for a long time despite all efforts. Hoáng et al. proved in [2] that k -COLORING can be solved in polynomial time on P_5 -free graphs for every positive integer k , that is the problem is in XP when parameterized by k . This leads to the question whether k -COLORING is FPT on P_5 -free graphs when parameterized by k . The question about complexity of k -COLORING is also open and interesting for $2K_2$ -free graphs that compose a proper subclass of P_5 -free graphs.

References

- 1 P. A. GOLOVACH, M. JOHNSON, D. PAULUSMA, AND J. SONG, *A survey on the computational complexity of colouring graphs with forbidden subgraphs*, J. Graph Theory, 84 (2017), pp. 331–363.
- 2 C. T. HOÀNG, M. KAMINSKI, V. V. LOZIN, J. SAWADA, AND X. SHU, *Deciding k -colorability of P_5 -free graphs in polynomial time*, Algorithmica, 57 (2010), pp. 74–81.

4.3 Finding H -Induced Minors

Daniel Paulusma (Durham University, GB)

License  Creative Commons BY 4.0 International license
© Daniel Paulusma

A graph G contains a graph H as an induced minor if H can be obtained from G by a sequence of vertex deletions and edge contractions. For a fixed graph H (that is, H is not part of the input), the H -Induced Minor problem is to decide if a given graph G contains H as an induced minor.

The complexity of H -Induced Minor has been partially classified by Fellows et al. [1]), but the classification is open even for trees. The smallest open case for trees [2] is the case where H is the tree H^* that is obtained from subdividing the centre edge of a double star with two leaves on each side. That is, H is the graph with vertices $a_1, a_2, b, c, d, e_1, e_2$ and edges $a_1b, a_2b, bc, cd, de_1, e_1e_2$.


What is the complexity of H^* -Induced Minor?

References

- 1 M.R. Fellows, J. Kratochvíl, M. Middendorf, and F. Pfeiffer. The Complexity of Induced Minors and Related Problems. *Algorithmica* 13:266–282 (1995)
- 2 J. Fiala, M. Kamiński and D. Paulusma. Detecting induced star-like minors in polynomial time. *Journal of Discrete Algorithms* 17:74–85 (2012)

4.4 k -Community

Jan Arne Telle (University of Bergen, NO)

License  Creative Commons BY 4.0 International license
© Jan Arne Telle

A partition C_1, C_2, \dots, C_k of the vertex set of a graph is a k -community, for $k \geq 2$, if for all i we have $|C_i| \geq 2$ and for all $i \neq j$ and any $v \in C_i$ we have $\frac{|N(v) \cap C_i|}{|C_i|} \geq \frac{|N(v) \cap C_j|}{|C_j|}$. As a cocktail party problem, we can view it as an organization of guests into groups so that everyone knows a higher percentage of people in their own group than they know in any other group. In 2013 Martin Olsen (Math. Social Sciences) showed that every graph on at least 4 vertices except the star graphs have a k -community for some value of $k \geq 2$. However, the question of deciding for a fixed k , if a graph allows a k -community, is not known to be solvable in polynomial time, and neither is it known to be NP-complete, for any $k \geq 2$. Most work has been done on the case $k = 2$. See the 2018 paper by Bazgan et al, Structural and Algorithmic Properties of 2-Community Structures (*Algorithmica*) for more on this.

Participants

- Tara Abrishami
Princeton University, US
- Akanksha Agrawal
Indian Institute of Technology
Madras, IN
- Bogdan Alecu
University of Leeds, GB
- Christoph Brause
TU Bergakademie Freiberg, DE
- Nick Brettell
Victoria University –
Wellington, NZ
- Henning Bruhn-Fujimoto
Universität Ulm, DE
- Maria Chudnovsky
Princeton University, US
- Konrad Dabrowski
Newcastle University, GB
- Peter Gartland
University of California –
Santa Barbara, US
- Jan Goedgebeur
KU Leuven, BE
- Petr A. Golovach
University of Bergen, NO
- Bart Jansen
TU Eindhoven, NL
- Tuukka Korhonen
University of Bergen, NO
- Daniel Král’
Masaryk University – Brno, CZ
- Madhumita Kundu
University of Bergen, NO
- Paloma Lima
IT University of
Copenhagen, DK
- Daniel Lokshtanov
University of California –
Santa Barbara, US
- Barnaby Martin
Durham University, GB
- Tomas Masarik
University of Warsaw, PL
- Jana Masarikova
University of Warsaw, PL &
Charles University – Praha, CZ
- Andrea Munaro
Queen’s University of
Belfast, GB
- Jelle Oostveen
Utrecht University, NL
- Sukanya Pandey
Utrecht University, NL
- Daniel Paulusma
Durham University, GB
- Marcin Pilipczuk
University of Warsaw, PL
- Michal Pilipczuk
University of Warsaw, PL
- Pawel Rzazewski
Warsaw University of
Technology, PL
- Saket Saurabh
The Institute of Mathematical
Sciences – Chennai, IN
- Oliver Schaudt
Bayer AG – Leverkusen, DE
- Paul Seymour
Princeton University, US
- Roohani Sharma
MPI für Informatik –
Saarbrücken, DE
- Siani Smith
University of Bristol, GB
- Jan Arne Telle
University of Bergen, NO
- Nicolas Trotignon
ENS – Lyon, FR
- Erik Jan van Leeuwen
Utrecht University, NL
- Kristina Vuskovic
University of Leeds, GB



Counting and Sampling: Algorithms and Complexity

Holger Dell^{*1}, Mark R. Jerrum^{*2}, Haiko Müller^{*3}, Konrad Anand^{†4},
and Marcus Pappik^{†5}

1 Goethe-Universität – Frankfurt am Main, DE. dagstuhl.de@holgerdell.com

2 Queen Mary University of London, GB. m.jerrum@qmul.ac.uk

3 University of Leeds, GB. h.muller@leeds.ac.uk

4 Queen Mary University of London, G. k.anand@qmul.ac.uk

5 Hasso-Plattner-Institut, Universität Potsdam, DE. marcus.pappik@hpi.de

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 22482 “Counting and Sampling: Algorithms and Complexity”. We document the talks presented, covering many advances in the area made over the last five years. As well, we document the progress made by working groups on future projects.

Seminar November 27–2, 2022 – <http://www.dagstuhl.de/22482>

2012 ACM Subject Classification Theory of computation → Computational complexity and cryptography; Theory of computation → Design and analysis of algorithms; Theory of computation → Randomness, geometry and discrete structures; Mathematics of computing → Discrete mathematics; Mathematics of computing → Probability and statistics; Mathematics of computing → Mathematical analysis

Keywords and phrases Sampling, Counting, Algorithms, Complexity, Statistical Physics, Phase Transitions, Markov Chains, Graphs, Point Processes

Digital Object Identifier 10.4230/DagRep.12.11.124

1 Executive Summary

Holger Dell (Goethe-Universität – Frankfurt am Main, DE)

Mark R. Jerrum (Queen Mary University of London, GB)

Haiko Müller (University of Leeds, GB)

License  Creative Commons BY 4.0 International license
 Holger Dell, Mark R. Jerrum, and Haiko Müller

Counting and sampling problems arise in areas such as statistics (benchmarking statistical tests, or sampling from a posterior distribution) and statistical physics (computing the partition function of a spin system). Computationally, these problems are very different in character from decision or optimisation problems, and their solution requires distinctive techniques. It is natural to treat counting and sampling together in the same Dagstuhl Seminar, as they are closely related computationally: subject to a reasonable side condition, an efficient algorithm for sampling certain combinatorial structures can be used as a black box to approximately count those structures, and vice versa.

Although much attention has been directed towards the complexity of counting and sampling problems, our understanding of them is not as well developed as it is of decision and optimisation problems. This Seminar marks a timely return to the topic, as new ideas have recently been injected into the area, resulting in renewed activity and progress. It is

* Editor / Organizer

† Editorial Assistant / Collector



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

Counting and Sampling: Algorithms and Complexity, *Dagstuhl Reports*, Vol. 12, Issue 11, pp. 124–145
Editors: Holger Dell, Mark R. Jerrum, Haiko Müller, Konrad Anand, and Marcus Pappik



Dagstuhl Reports

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

particularly satisfying to observe that much of this progress has been in the positive direction, in the form of new efficient algorithms. This is in an area where negative results had become the norm.

The Covid pandemic inevitably left its mark on the meeting. Over five years elapsed between the previous Dagstuhl Seminar on a related topic and the current one. In the meantime, the introduction of a circle of ideas around high-dimensional expanders, spectral expansion and entropy decay has transformed the analysis of Markov chains for sampling, and brought many previously intractable questions within scope of our methods. An unwelcome impact of Covid was to reduce significantly the number of participants. Sadly, it was not possible to invite all the people we would have liked to see at the meeting.

With a view to providing a snapshot of current interests, here is a rough-and-ready breakdown of the presentations against a somewhat arbitrary set of headings.

- *Connections with statistical physics, phase transitions, etc.* Coja-Oghlan, Galanis and Patel,
- *Holant and constraint satisfaction problems.* Backens and Bulatov.
- *Markov chains.* Guo and Miracle.
- *Parameterised complexity of counting problems.* Bressan, Focke, Roth and Wellnitz.
- *Perfect samplers.* Anand and Cannon.
- *Point processes and other geometric connections.* Anari, Jerrum and Pappik.
- *Polynomials associated with graphs, matroids and matrices.* Björklund, Curticapean, Regts.
- *Other.* Göbel, Goldberg, Kaski, Lapinskas.

If nothing else, this rough classification exercise gives an impression of the wide span of current research. Aside from the progress on the analysis of Markov chains mentioned earlier, many other topics have seen advances in the past five years. Examples include: counting small patterns (‘motifs’) in large graphs (networks), sampling structures in regions of phase non-uniqueness, perfect sampling, and weighted counting problems where the weights are complex. It turns out that the latter study shines light on the case of real weights, through an examination of zeros of partition functions in the complex plane. The meeting gave participants a long-awaited chance to review developments over the past five years.

On the organisational front, an innovation (as far as this community is concerned) was the inclusion of a problem session on the first day. This went off quite smoothly, and small working groups formed fairly spontaneously to work on problems during the week. On the final day we heard from the groups a summary of their investigations over the week. Our hope is that sufficient momentum was achieved on some of these problems that groups will continue to work on them beyond the end of the meeting. Indeed, one of the working groups decided to apply to run a workshop on homomorphism counting at ICALP 2023 with this aim in mind. The proposal, by Radu Curticapean and Marc Roth, was accepted, and the workshop, entitled “ADjoint HOMomorphism Counting” (AD HOC) will take place in July 2023. We look forward to being able to report on advances achieved on this and other topics on this website.

2 Table of Contents

Executive Summary

Holger Dell, Mark R. Jerrum, and Haiko Müller 124

Overview of Talks

Lazy Depth-First Sampling of Spin Systems
Konrad Anand and Mark R. Jerrum 128

Parallel Discrete Sampling via Continuous Walks
Nima Anari 128

Holant clones and approximation of holant problems
Miriam Backens 129

The Fine-Grained Complexity of Computing the Tutte Polynomial of a Linear Matroid
Andreas Björklund 129

Linear and sublinear algorithms for sampling graphlets in large graphs
Marco Bressan 129

Complexity classification of counting graph homomorphisms modulo a prime number
Andrei A. Bulatov 130

Fast and Perfect Sampling of Subgraphs and Polymer Systems
Sarah Cannon 130

The random 2-SAT partition function
Amin Coja-Oghlan and Noela Müller 131

Immanants and determinants
Radu Curticapean 131

Counting small induced subgraphs with hereditary properties
Jacob Focke 132

Metastability for the ferromagnetic Potts model
Andreas Galanis 132

Instability of contention resolution protocols
Leslie Ann Goldberg 133

Towards derandomising Markov chain Monte Carlo
Heng Guo 133

Analysis of the survival time of the SIRS process via expansion
Andreas Göbel and Marcus Pappik 134

Counting vertices of integral polytopes defined by facets
Mark R. Jerrum 135

Nearly optimal independence oracle algorithms for edge estimation in hypergraphs
John Lapinskas 135

Iterated Decomposition of Biased Permutations Via New Bounds on the Spectral Gap of Markov Chains
Sarah Miracle 136

Discretization-based algorithms for repulsive Gibbs point processes <i>Marcus Pappikrich and Andreas Göbel</i>	136
Sampling from the low temperature ferromagnetic Potts model via flows <i>Viresh Patel and Guus Regts</i>	137
Trustworthy Monte Carlo <i>Petteri Kaski</i>	137
Approximating the chromatic polynomial is as hard as computing it exactly <i>Guus Regts</i>	137
Counting Small Directed Subgraphs, Parameterised by the Outdegree <i>Marc Roth</i>	138
Tight Complexity Bounds for Counting Generalized Dominating Sets in Bounded-Treewidth Graphs <i>Philip Wellnitz</i>	138
Working groups	
Counting Functions via Extension Oracles <i>Marco Bressan, Konrad Anand, and Holger Dell</i>	139
Independent sets of fixed size <i>Ewan Davies, Sarah Cannon, Charlie Carlson, Sarah Miracle, and Noela Müller</i>	140
Fine grained complexity of counting independent sets in bounded degree graphs <i>Heng Guo</i>	141
Approximating the number of proper colorings of a planar graph with a large number of colors <i>Viresh Patel, Andrei A. Bulatov, Charlie Carlson, Heng Guo, and Guus Regts</i>	142
Understanding the Homomorphism Basis <i>Marc Roth, Marco Bressan, Radu Curticapean, Holger Dell, Jacob Focke, and Philip Wellnitz</i>	142
Participants	145

3 Overview of Talks

3.1 Lazy Depth-First Sampling of Spin Systems

Konrad Anand (Queen Mary University of London, GB) and Mark R. Jerrum (Queen Mary University of London, GB)

License © Creative Commons BY 4.0 International license

© Konrad Anand and Mark R. Jerrum

Main reference Konrad Anand, Mark Jerrum: “Perfect Sampling in Infinite Spin Systems Via Strong Spatial Mixing”, *SIAM Journal on Computing*, Vol. 51(4), pp. 1280–1295, 2022.

URL <https://doi.org/10.1137/21M1437433>

We present a simple algorithm that perfectly samples configurations from the unique Gibbs measure of a spin system on a potentially infinite graph G . The sampling algorithm assumes strong spatial mixing together with subexponential growth of G . It produces a finite window onto a perfect sample from the Gibbs distribution. The run-time is linear in the size of the window, in particular it is constant for each vertex.

3.2 Parallel Discrete Sampling via Continuous Walks

Nima Anari (Stanford University, US)

License © Creative Commons BY 4.0 International license

© Nima Anari

Joint work of Nima Anari, Yizhi Huang, Tianyu Liu, Thuy-Duong Vuong, Brian Xu, Katherine Yu

We develop a framework for sampling from discrete distributions on the hypercube by sampling from continuous distributions obtained by convolution with spherical Gaussians. We show that for well-studied families of discrete distributions, the result of the convolution is well-conditioned log-concave, whenever the Gaussian’s variance is above an $O(1)$ threshold. We plug in off-the-shelf continuous sampling methods into our framework to obtain novel discrete sampling algorithms. Additionally, we introduce and study a crucial notion of smoothness for discrete distributions that we call transport-stability, that we use to control the propagation of error in our framework. We expect transport-stability to be of independent interest, as we connect it to constructions of optimally mixing local random walks and concentration inequalities.

As our main application, we resolve open questions on parallel sampling of distributions which admit parallel counting. We show that determinantal point processes can be sampled via RNC algorithms, that is in time $\text{polylog}(n)$ using $\text{poly}(n)$ processors. For a wider class of distributions, we show our framework yields Quasi-RNC sampling, that is sampling in time $\text{polylog}(n)$ using $n^{O(\log n)}$ processors. This wider class includes random Eulerian tours in digraphs.

3.3 Holant clones and approximation of holant problems

Miriam Backens (University of Birmingham, GB)

License © Creative Commons BY 4.0 International license
© Miriam Backens

Joint work of Miriam Backens, Leslie Ann Goldberg

Main reference Miriam Backens, Leslie Ann Goldberg: “Holant Clones and the Approximability of Conservative Holant Problems”, *ACM Trans. Algorithms*, Vol. 16(2), pp. 23:1–23:55, 2020.

URL <https://doi.org/10.1145/3381425>

Holant problems are a generalisation of counting constraint satisfaction problems, equivalent to the problem of fully contracting a tensor network built from some fixed family of tensors. Generalising relational and functional clones, the holant clone of a set of constraint functions contains all functions that can be simulated from the original set via gadgets.

I will discuss a result about approximation of holant problems that employs the formalism of holant clones, and talk about some further work in progress.

3.4 The Fine-Grained Complexity of Computing the Tutte Polynomial of a Linear Matroid

Andreas Björklund (Lund, SE)

License © Creative Commons BY 4.0 International license
© Andreas Björklund

Joint work of Andreas Björklund and Petteri Kaski

Main reference Andreas Björklund, Petteri Kaski: The Fine-Grained Complexity of Computing the Tutte Polynomial of a Linear Matroid. *SODA 2021*: 2333–2345.

URL <https://dl.acm.org/doi/10.5555/3458064.3458203>

We show that computing the Tutte polynomial of a linear matroid of dimension k on $k^{O(1)}$ points over a field of $k^{O(1)}$ elements requires $k^{\Omega(k)}$ time unless the #ETH— a counting extension of the Exponential Time Hypothesis of Impagliazzo and Paturi [CCC 1999] due to Dell et al. [ACM TALG 2014]—is false.

3.5 Linear and sublinear algorithms for sampling graphlets in large graphs

Marco Bressan (University of Milan, IT)

License © Creative Commons BY 4.0 International license
© Marco Bressan

Main reference Marco Bressan: “Efficient and near-optimal algorithms for sampling connected subgraphs”, in Proc. of the STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21–25, 2021, pp. 1132–1143, ACM, 2021.

URL <https://doi.org/10.1145/3406325.3451042>

A fundamental primitive in modern graph mining is sampling connected subgraphs on k vertices (also known as k -graphlets) from a graph G . For a long time, no good algorithm was known for sampling k -graphlets uniformly at random; the best algorithms available could sample only approximately. In this talk I will present algorithms for sampling k -graphlets uniformly or eps-uniformly from an arbitrary n -vertex graph G with preprocessing time linear or even sublinear in G and sampling time logarithmic or even constant in G .

3.6 Complexity classification of counting graph homomorphisms modulo a prime number

Andrei A. Bulatov (Simon Fraser University – Burnaby, CA)

License © Creative Commons BY 4.0 International license
© Andrei A. Bulatov

Joint work of Andrei A. Bulatov, Amirhosein Kazemina

Counting graph homomorphisms and its generalizations such as the Counting Constraint Satisfaction Problem (CSP), its variations, and counting problems in general have been intensively studied since the pioneering work of Valiant. While the complexity of exact counting of graph homomorphisms (Dyer and Greenhill, 2000) and the counting CSP (Bulatov, 2013, and Dyer and Richerby, 2013) is well understood, counting modulo some natural number has attracted considerable interest as well. In their 2015 paper Faben and Jerrum suggested a conjecture stating that counting homomorphisms to a fixed graph H modulo a prime number is hard whenever it is hard to count exactly, unless H has automorphisms of certain kind. In this paper we confirm this conjecture. As a part of this investigation we develop techniques that widen the spectrum of reductions available for modular counting and apply to the general CSP rather than being limited to graph homomorphisms.

3.7 Fast and Perfect Sampling of Subgraphs and Polymer Systems

Sarah Cannon (Claremont McKenna College, US)

License © Creative Commons BY 4.0 International license
© Sarah Cannon

Joint work of Antonio Blanca, Sarah Cannon, Will Perkins

Main reference Antonio Blanca, Sarah Cannon, Will Perkins: “Fast and Perfect Sampling of Subgraphs and Polymer Systems”, in Proc. of the Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2022, September 19-21, 2022, University of Illinois, Urbana-Champaign, USA (Virtual Conference), LIPIcs, Vol. 245, pp. 4:1–4:18, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

URL <https://doi.org/10.4230/LIPIcs.APPROX/RANDOM.2022.4>

We give an efficient perfect sampling algorithm for weighted, connected induced subgraphs (or graphlets) of rooted, bounded degree graphs. Our algorithm utilizes a vertex-percolation process with a carefully chosen rejection filter and works under a percolation subcriticality condition. We show that this condition is optimal in the sense that the task of (approximately) sampling weighted rooted graphlets becomes impossible in finite expected time for infinite graphs and intractable for finite graphs when the condition does not hold. We apply our sampling algorithm as a subroutine to give near linear-time perfect sampling algorithms for polymer models and weighted non-rooted graphlets in finite graphs, two widely studied yet very different problems. This new perfect sampling algorithm for polymer models gives improved sampling algorithms for spin systems at low temperatures on expander graphs and unbalanced bipartite graphs, among other applications.

3.8 The random 2-SAT partition function

Amin Coja-Oghlan (TU Dortmund, DE) and Noela Müller (TU Eindhoven, NL)

License © Creative Commons BY 4.0 International license
© Amin Coja-Oghlan and Noela Müller

Joint work of Amin Coja-Oghlan, Dimitris Achlioptas, Max Hahn-Klimroth, Joon Lee, Noëla Müller, Manuel Penschuck, Guangyan Zhou

Main reference Dimitris Achlioptas, Amin Coja-Oghlan, Max Hahn-Klimroth, Joon Lee, Noëla Müller, Manuel Penschuck, Guangyan Zhou: “The number of satisfying assignments of random 2-SAT formulas”, *Random Struct. Algorithms*, Vol. 58(4), pp. 609–647, 2021.

URL <https://doi.org/10.1002/rsa.20993>

The random 2-SAT problem was the first random constraint satisfaction problem whose satisfiability threshold could be pinpointed precisely [1, 2]. The satisfiability threshold turns out to be determined by the appearance of certain local structures called “bicycles”. In this talk I address the more difficult but no less fundamental problem of calculating the (exponential order of the) number of satisfying assignments within the satisfiable phase. The main result rigorously establishes a prediction from statistical physics called the “replica symmetric ansatz” [3, 4]. The resulting formula does not boil down to a simple algebraic expression, but rather involves a stochastic fixed point problem that mimics the Belief Propagation message passing algorithm. Nonetheless, the formula can be evaluated numerically within arbitrary precision.

References

- 1 V. Chvatal, B. Reed: Mick gets some (the odds are on his side). *Proc. 33th FOCS* (1992) 620–627.
- 2 A. Goerdt: A threshold for unsatisfiability. *J. Comput. Syst. Sci.* **53** (1996) 469–486
- 3 R. Monasson, R. Zecchina: The entropy of the k -satisfiability problem. *Phys. Rev. Lett.* **76** (1996) 3881.
- 4 R. Monasson, R. Zecchina: Statistical mechanics of the random K -SAT model. *Phys. Rev. E* **56** (1997) 1357–1370.

3.9 Immanants and determinants

Radu Curticapean (IT University of Copenhagen, DK)

License © Creative Commons BY 4.0 International license
© Radu Curticapean

Main reference Radu Curticapean: “A full complexity dichotomy for immanant families”, in *Proc. of the STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21–25, 2021*, pp. 1770–1783, ACM, 2021.

URL <https://doi.org/10.1145/3406325.3451124>

Immanants are matrix functionals that generalize determinants and permanents by allowing general irreducible characters of the symmetric group as permutations weights rather than merely the sign function (which yields the determinant) or the all-ones function (which yields the permanent). In this talk, we give an introduction to immanants and describe a recent classification of their complexity. In a second part, we give a simple proof that shows how determinants can be expressed in terms of homomorphism counts from cycle covers, leaving open similar expressions for general immanants.

3.10 Counting small induced subgraphs with hereditary properties

Jacob Focke (CISPA – Saarbrücken, DE)

License  Creative Commons BY 4.0 International license
© Jacob Focke

Joint work of Jacob Focke, Marc Roth

Main reference Jacob Focke, Marc Roth: “Counting small induced subgraphs with hereditary properties”, in Proc. of the STOC ’22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 – 24, 2022, pp. 1543–1551, ACM, 2022.

URL <https://doi.org/10.1145/3519935.3520008>

We study the computational complexity of the problem $\#\text{IndSub}(\Phi)$ of counting k -vertex induced subgraphs of a graph G that satisfy a graph property Φ . Our main result establishes an exhaustive and explicit classification for all hereditary properties, including tight conditional lower bounds under the Exponential Time Hypothesis (ETH): If a hereditary property Φ is true for all graphs, or if it is true only for finitely many graphs, then $\#\text{IndSub}(\Phi)$ is solvable in polynomial time. Otherwise, $\#\text{IndSub}(\Phi)$ is $\#\text{W}[1]$ -complete when parameterised by k , and, assuming ETH, it cannot be solved in time $f(k) \cdot |G|^{\sigma(k)}$ for any function f . This classification features a wide range of properties for which the corresponding detection problem (as classified by Khot and Raman [TCS 02]) is tractable but counting is hard. Moreover, even for properties which are already intractable in their decision version, our results yield significantly stronger lower bounds for the counting problem. As additional result, we also present an exhaustive and explicit parameterised complexity classification for all properties that are invariant under homomorphic equivalence. By covering one of the most natural and general notions of closure, namely, closure under vertex-deletion (hereditary), we generalise some of the earlier results on this problem. For instance, our results fully subsume and strengthen the existing classification of $\#\text{IndSub}(\Phi)$ for monotone (subgraph-closed) properties due to Roth, Schmitt, and Wellnitz [FOCS 20]. A full version of our paper, containing all proofs, is available at <https://arxiv.org/abs/2111.02277>.

3.11 Metastability for the ferromagnetic Potts model

Andreas Galanis (University of Oxford, GB)

License  Creative Commons BY 4.0 International license
© Andreas Galanis

Joint work of Amin Coja-Oghlan, Andreas Galanis, Leslie Ann Goldberg, Jean Bernoulli Ravelomanana, Daniel Stefankovic, Eric Vigoda

Main reference Amin Coja-Oghlan, Andreas Galanis, Leslie Ann Goldberg, Jean Bernoulli Ravelomanana, Daniel Stefankovic, Eric Vigoda: “Metastability of the Potts ferromagnet on random regular graphs”, CoRR, Vol. abs/2202.05777, 2022.

URL <https://arxiv.org/abs/2202.05777>

We study the performance of Markov chains for the q -state ferromagnetic Potts model on random regular graphs. While the cases of the grid and the complete graph are by now well-understood, the case of random regular graphs has resisted a detailed analysis and, in fact, even analysing the properties of the Potts distribution has remained elusive. It is conjectured that the performance of Markov chains is dictated by metastability phenomena, i.e., the presence of “phases” (clusters) in the sample space where Markov chains with local update rules, such as the Glauber dynamics, are bound to take exponential time to escape, and therefore cause slow mixing. The phases that are believed to drive these metastability phenomena in the case of the Potts model emerge as local, rather than global, maxima of the so-called Bethe functional, and previous approaches of analysing these phases based on optimisation arguments fall short of the task.

Our first contribution is to detail the emergence of the two relevant phases for the q -state Potts model on the d -regular random graph for all integers $q, d \geq 3$, and establish that for an interval of temperatures, delineated by the uniqueness and a broadcasting threshold on the d -regular tree, the two phases coexist (as possible metastable states). The proofs are based on a conceptual connection between spatial properties and the structure of the Potts distribution on the random regular graph, rather than complicated moment calculations. This significantly refines earlier results by Helmuth, Jenssen, and Perkins who had established phase coexistence for a small interval around the so-called ordered-disordered threshold (via different arguments) that applied for large q and $d \geq 5$.

Based on our new structural understanding of the model, our second contribution is to obtain metastability results for two classical Markov chains for the Potts model. We first complement recent fast mixing results for Glauber dynamics by Blanca and Gheissari below the uniqueness threshold, by showing an exponential lower bound on the mixing time above the uniqueness threshold. Then, we obtain tight results even for the non-local and more elaborate Swendsen-Wang chain, where we establish slow mixing/metastability for the whole interval of temperatures where the chain is conjectured to mix slowly on the random regular graph. The key is to bound the conductance of the chains using a random graph “planting” argument combined with delicate bounds on random-graph percolation.

3.12 Instability of contention resolution protocols

Leslie Ann Goldberg (University of Oxford, GB)

License © Creative Commons BY 4.0 International license
© Leslie Ann Goldberg

Joint work of Leslie Ann Goldberg, John Lapinskas

Main reference Leslie Ann Goldberg, John Lapinskas: “Instability of backoff protocols with arbitrary arrival rates”, CoRR, Vol. abs/2203.17144, 2022.

URL <https://doi.org/10.48550/arXiv.2203.17144>

A backoff protocol is a simple and elegant randomised algorithm for communicating in a Multiple Access Channel. Aldous conjectured in 1987 that, for any positive arrival rate, every backoff protocol is unstable. I will report on new work with John Lapinskas towards proving this conjecture. (This will appear in SODA 2023.)

3.13 Towards derandomising Markov chain Monte Carlo

Heng Guo (University of Edinburgh, GB)

License © Creative Commons BY 4.0 International license
© Heng Guo

Joint work of Heng Guo, Weiming Feng, Chunyang Wang, Jiaheng Wang, Yitong Yin

Main reference Weiming Feng, Heng Guo, Chunyang Wang, Jiaheng Wang, Yitong Yin: “Towards derandomising Markov chain Monte Carlo”, CoRR, Vol. abs/2211.03487, 2022.

URL <https://doi.org/10.48550/arXiv.2211.03487>

We present a new framework to derandomise certain Markov chain Monte Carlo (MCMC) algorithms. As in MCMC, we first reduce counting problems to sampling from a sequence of marginal distributions. For the latter task, we introduce a method called coupling towards the past that can, in logarithmic time, evaluate one or a constant number of variables from a stationary Markov chain state. Since there are at most logarithmic random choices, this leads

to very simple derandomisation. We provide two applications of this framework, namely efficient deterministic approximate counting algorithms for hypergraph independent sets and hypergraph colourings, under local lemma type conditions matching, up to lower order factors, their state-of-the-art randomised counterparts.

3.14 Analysis of the survival time of the SIRS process via expansion

Andreas Göbel (Hasso-Plattner-Institut, Universität Potsdam, DE) and Marcus Pappik (Hasso-Plattner-Institut, Universität Potsdam, DE)

License © Creative Commons BY 4.0 International license

© Andreas Göbel and Marcus Pappik

Joint work of Tobias Friedrich, Andreas Göbel, Nicolas Klodt, Martin S. Krejca, Marcus Pappik

Main reference Tobias Friedrich, Andreas Göbel, Nicolas Klodt, Martin S. Krejca, Marcus Pappik: “Analysis of the survival time of the SIRS process via expansion”, arXiv, 2022.

URL <https://doi.org/10.48550/ARXIV.2205.02653>

We study the SIRS process, a continuous-time Markov chain modelling the spread of infections on graphs. In this process, vertices are either susceptible, infected, or recovered. Each infected vertex becomes recovered at rate 1 and infects each of its susceptible neighbours independently at rate λ , and each recovered vertex becomes susceptible at a rate ρ , which we assume to be independent of the graph size. A central quantity of the SIRS process is the time until no vertex is infected, known as the *survival time*. The survival time of the SIRS process is studied extensively in a variety of contexts. Surprisingly though, to the best of our knowledge, no rigorous theoretical results exist so far. This is even more surprising given that for the related SIS process, mathematical analysis began in the 70s and continues to this day.

We address this imbalance by conducting the first theoretical analyses of the SIRS process on various graph classes via their expansion properties. Our analyses assume that the graphs start with at least one infected vertex and no recovered vertices. Our first result considers stars, which have poor expansion. We prove that the expected survival time of the SIRS process on stars is at most polynomial in the graph size for *any* value of λ . This behaviour is fundamentally different from the SIS process, where the expected survival time is exponential already for small infection rates. Due to this property, for the SIS process, stars constitute an important sub-structure for proving an expected exponential survival time of more complicated graphs. For the SIRS process, this argument is not sufficient.

Our main result is an exponential lower bound of the expected survival time of the SIRS process on expander graphs. Specifically, we show that on expander graphs G with n vertices, degree close to d , and sufficiently small spectral expansion, the SIRS process has expected survival time at least exponential in n when $\lambda \geq c/d$ for a constant $c > 1$. This result is complemented by established results for the SIS process, which imply that the expected survival time of the SIRS process is at most logarithmic in n when $\lambda \leq c/d$ for a constant $c < 1$. Combined, our result shows an almost-tight threshold behaviour of the expected survival time of the SIRS process on expander graphs. Additionally, our result holds even if G is a subgraph. This allows, for the SIRS process, the use of expanders as sub-structures for lower bounds, similar to stars in the SIS process. Notably, our result implies an almost-tight threshold for Erdős–Rényi graphs and a regime of exponential survival time for hyperbolic random graphs, one of the most popular graph models, as it incorporates many properties found in real-world networks. The proof of our main result draws inspiration from Lyapunov functions used in mean-field theory to devise a two-dimensional potential function and applying a negative-drift theorem to show that the expected survival time is exponential.

3.15 Counting vertices of integral polytopes defined by facets

Mark R. Jerrum (Queen Mary University of London, GB)

License © Creative Commons BY 4.0 International license
© Mark R. Jerrum

Joint work of Heng Guo (Edinburgh) and Mark Jerrum

Main reference Heng Guo, Mark Jerrum: “Counting vertices of integral polytopes defined by facets”, *Discrete and Computational Geometry*, July 2022.

URL <https://doi.org/10.1007/s00454-022-00406-8>

We present a number of complexity results concerning the problem of counting vertices of an integral polytope defined by a system of linear inequalities. The focus is on polytopes with small integer vertices, particularly 0/1-polytopes and half-integral polytopes (ones whose vertices are contained in $\{0, 1\}^n$ and $\{0, \frac{1}{2}, 1\}^n$, respectively). Such polytopes are ubiquitous in the field of combinatorial optimisation.

Suppose a polytope P is defined by linear inequalities $Ax \leq b$. If the matrix A is ‘totally unimodular’ then P is guaranteed to be integral; many integral polytopes that are encountered in practice arise in this way. Network matrices and their transposes are particular kinds of totally unimodular matrices. Our main results are the following.

- Counting the vertices of a 0/1-polytope exactly is #P complete, even when restricted to the case when A is a network matrix or the transpose of one. (This is nothing more than an observation, given that the problems of counting perfect matchings or independent sets in a bipartite graph are both #P-complete.)
- Approximately counting the vertices of a half-integral polytope is NP-hard, as witnessed by the ‘perfect 2-matching polytope’.
- The vertex-counting problem for 0/1-polytopes defined by transposes of network matrices is equivalent, under approximation-preserving reductions, to counting independent sets of a bipartite graph (#BIS). No efficient approximation algorithm is known for #BIS, but neither is approximating #BIS known to be NP-hard. Many natural counting problems are known to be equivalent to #BIS under polynomial-time approximation-preserving reductions.
- For a natural subclass of polytopes defined by network matrices, it is possible to approximate the number of vertices in polynomial time. The complexity for network matrices in general is, however, unknown,

3.16 Nearly optimal independence oracle algorithms for edge estimation in hypergraphs

John Lapinskas (University of Bristol, GB)

License © Creative Commons BY 4.0 International license
© John Lapinskas

Joint work of John Lapinskas, Holger Dell, Kitty Meeks

Main reference Holger Dell, John Lapinskas, Kitty Meeks: “Nearly optimal independence oracle algorithms for edge estimation in hypergraphs”, *CoRR*, Vol. abs/2211.03874, 2022.

URL <https://doi.org/10.48550/arXiv.2211.03874>

We study a query model of computation in which an n -vertex k -hypergraph can be accessed only via its independence oracle or via its colourful independence oracle, and each oracle query may incur a cost depending on the size of the query. In each of these models, we obtain oracle algorithms to approximately count the hypergraph’s edges, and we unconditionally prove that no oracle algorithm for this problem can have significantly smaller worst-case oracle cost than our algorithms.

3.17 Iterated Decomposition of Biased Permutations Via New Bounds on the Spectral Gap of Markov Chains

Sarah Miracle (University of St. Thomas – St. Paul, US)

License © Creative Commons BY 4.0 International license
© Sarah Miracle

Joint work of Sarah Miracle, Amanda Pascoe Streib, Noah Streib

Main reference Sarah Miracle, Amanda Pascoe Streib, Noah Streib: “Iterated Decomposition of Biased Permutations via New Bounds on the Spectral Gap of Markov Chains”, in Proc. of the Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2020, August 17-19, 2020, Virtual Conference, LIPIcs, Vol. 176, pp. 3:1–3:21, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020.

URL <https://doi.org/10.4230/LIPIcs.APPROX/RANDOM.2020.3>

We study a nearest-neighbor Markov chain over biased permutations of $[n]$. We build on previous work that analyzed the spectral gap of the chain when $[n]$ is partitioned into k classes. There, the authors iteratively decomposed the nearest neighbor chain into simpler chains, but incurred a multiplicative penalty of n^{-2} for each application of the decomposition theorem. We introduce a new decomposition theorem which allows us to avoid this penalty (in certain cases) and prove the first inverse-polynomial bound on the spectral gap of the chain when k is as large as $\Theta(n/\log n)$. The previous best known bound assumed k was at most a constant.

3.18 Discretization-based algorithms for repulsive Gibbs point processes

Marcus Pappik (Hasso-Plattner-Institut, Universität Potsdam, DE) and Andreas Göbel (Hasso-Plattner-Institut, Universität Potsdam, DE)

License © Creative Commons BY 4.0 International license
© Marcus Pappik and Andreas Göbel

Joint work of Marcus Pappik, Tobias Friedrich, Andreas Göbel, Maximilian Katzmann, and Martin S. Krejca

Main reference Tobias Friedrich, Andreas Göbel, Maximilian Katzmann, Martin S. Krejca, Marcus Pappik: “Using random graphs to sample repulsive Gibbs point processes with arbitrary-range potentials”, CoRR, Vol. abs/2204.01793, 2022.

URL <https://doi.org/10.48550/arXiv.2204.01793>

Gibbs point processes are a popular way to model particle distributions of fluids and gasses in Euclidean space. Similar to spin systems on graphs, which might be seen as their discrete counterparts, sampling the Gibbs distribution and computing its normalizing constant, the partition function, of such point processes is highly relevant. However, until recently, very few rigorous computational results existed. In this talk, we give a brief introduction to Gibbs point processes and present recent algorithmic results. In particular, we focus on discretization-based algorithms, which reduce the algorithmic tasks at hand to related problems for discrete spin systems, making use of the rich literature in that area. Our main focus will be on a recent approach that employs hard-core models on carefully constructed families of geometric random graphs to obtain sampling and approximation algorithms for Gibbs point processes. This results in efficient algorithms for arbitrary repulsive pair potentials ϕ up to a fugacity of $\lambda < e/C_\phi$, where C_ϕ is the temperedness constant of ϕ .

3.19 Sampling from the low temperature ferromagnetic Potts model via flows

Viresh Patel (Queen Mary University of London, GB) and Guus Regts (University of Amsterdam, NL)

License © Creative Commons BY 4.0 International license
© Viresh Patel and Guus Regts

Joint work of Viresh Patel, Jeroen Huijben, and Guus Regts

Main reference Jeroen Huijben, Viresh Patel, Guus Regts: “Sampling from the low temperature Potts model through a Markov chain on flows”, *Random Struct. Algorithms*, Vol. 62(1), pp. 219–239, 2023.

URL <https://doi.org/10.1002/rsa.21089>

I will discuss how one can (approximately and quickly) sample configurations from the ferromagnetic Potts model with underlying graph G at low temperatures using a Markov chain on flows. The use of flows allows one to work with certain types of graphs that can have unbounded degree.

3.20 Trustworthy Monte Carlo

Petteri Kaski

License © Creative Commons BY 4.0 International license
© Petteri Kaski

Joint work of Juha Harviainen, Mikko Koivisto, Petteri Kaski

Main reference Juha Harviainen, Mikko Koivisto, Petteri Kaski: “Trustworthy Monte Carlo”, in *Proc. of the Advances in Neural Information Processing Systems*, 2022.

URL <https://openreview.net/forum?id=jglXPY6gH-1>

Building on work of Williams (CCC’16) and Björklund & Kaski (PODC’16) on fine-grained noninteractive proof systems in the context of deterministic counting problems, we study verifiable randomized approximation schemes for hard counting problems such as the permanent. We show that sample-average-based Monte Carlo estimators such as the Godsil-Gutman and the Chien-Rasmussen-Sinclair estimators for the permanent admit verifiable randomized approximation with verifier complexity scaling essentially as the square root of the prover/estimator complexity.

[This work is to appear in NeurIPS’22.]

3.21 Approximating the chromatic polynomial is as hard as computing it exactly

Guus Regts (University of Amsterdam, NL)

License © Creative Commons BY 4.0 International license
© Guus Regts

Joint work of Guus Regts, Ferenc Bencs, Jeroen Huijben

Main reference Ferenc Bencs, Jeroen Huijben, Guus Regts: “Approximating the chromatic polynomial is as hard as computing it exactly”, *CoRR*, Vol. abs/2211.13790, 2022.

URL <https://doi.org/10.48550/arXiv.2211.13790>

In this talk I will explain that for any non-real algebraic number q , approximately computing the absolute value of the chromatic polynomial evaluated at q is as hard as computing it exactly and hence is #P-hard. The proof is based on constructing series-parallel gadgets that “implement” a dense set of edge interactions and is inspired by Sokal’s result saying that chromatic roots are dense in the complex plane.

3.22 Counting Small Directed Subgraphs, Parameterised by the Outdegree

Marc Roth (University of Oxford, GB)

License © Creative Commons BY 4.0 International license
© Marc Roth

Joint work of Marco Bressan, Matthias Lanzinger, Marc Roth

Main reference Marco Bressan, Matthias Lanzinger, Marc Roth: “The Complexity of Pattern Counting in Directed Graphs, Parameterised by the Outdegree”, CoRR, Vol. abs/2211.01905, 2022.

URL <https://doi.org/10.48550/arXiv.2211.01905>

We study the problem of counting the copies of a small directed pattern graph H in a large directed host graph G . Motivated by the recent surge on pattern counting in degenerate graphs, we focus on host graphs with small outdegree $d(G)$. Formally, we choose $|H| + d(G)$ as the problem parameter and ask for which classes of patterns the problem is fixed-parameter tractable.

This talk presents a complete parameterised complexity classification of the problem and provides an overview of the technical challenges in proving this result – among others, those challenges include a careful analysis of a variety of width measures on hypergraphs encoding the reachability structure of directed graphs.

3.23 Tight Complexity Bounds for Counting Generalized Dominating Sets in Bounded-Treewidth Graphs

Philip Wellnitz (MPI für Informatik – Saarbrücken, DE)

License © Creative Commons BY 4.0 International license
© Philip Wellnitz

Joint work of Jacob Focke, Dániel Marx, Fionn Mc Inerney, Daniel Neuen, Govind S. Sankar, Philipp Schepper, Philip Wellnitz

Main reference Jacob Focke, Dániel Marx, Fionn Mc Inerney, Daniel Neuen, Govind S. Sankar, Philipp Schepper, Philip Wellnitz: “Tight Complexity Bounds for Counting Generalized Dominating Sets in Bounded-Treewidth Graphs”, CoRR, Vol. abs/2211.04278, 2022.

URL <https://doi.org/10.48550/arXiv.2211.04278>

We investigate how efficiently a well-studied family of domination-type problems can be solved on bounded-treewidth graphs. For sets σ, ρ of non-negative integers, a (σ, ρ) -set of a graph G is a set S of vertices such that $|N(u) \cap S| \in \sigma$ for every $u \in S$, and $|N(v) \cap S| \in \rho$ for every $v \notin S$. The problem of finding a (σ, ρ) -set (of a certain size) unifies standard problems such as INDEPENDENT SET, DOMINATING SET, INDEPENDENT DOMINATING SET, and many others.

For all pairs of finite or cofinite sets (σ, ρ) , we determine (under standard complexity assumptions) the best possible value $c_{\sigma, \rho}$ such that there is an algorithm that counts (σ, ρ) -sets in time $c_{\sigma, \rho}^{\text{tw}} \cdot n^{O(1)}$ (if a tree decomposition of width tw is given in the input). Let s_{top} denote the largest element of σ if σ is finite, or the largest missing integer $+1$ if σ is cofinite; r_{top} is defined analogously for ρ . Surprisingly, $c_{\sigma, \rho}$ is often significantly smaller than the natural bound $s_{\text{top}} + r_{\text{top}} + 2$ achieved by existing algorithms [van Rooij, 2020]. Toward defining $c_{\sigma, \rho}$, we say that (σ, ρ) is m -structured if there is a pair (α, β) such that every integer in σ equals $\alpha \bmod m$, and every integer in ρ equals $\beta \bmod m$. Then, setting

- $c_{\sigma, \rho} = s_{\text{top}} + r_{\text{top}} + 2$ if (σ, ρ) is not m -structured for any $m \geq 2$,
- $c_{\sigma, \rho} = \max\{s_{\text{top}}, r_{\text{top}}\} + 2$ if (σ, ρ) is 2-structured, but not m -structured for any $m \geq 3$, and $s_{\text{top}} = r_{\text{top}}$ is even, and
- $c_{\sigma, \rho} = \max\{s_{\text{top}}, r_{\text{top}}\} + 1$, otherwise,

we provide algorithms counting (σ, ρ) -sets in time $c_{\sigma, \rho}^{\text{tw}} \cdot n^{O(1)}$. For example, for the EXACT INDEPENDENT DOMINATING SET problem (also known as PERFECT CODE) corresponding to $\sigma = \{0\}$ and $\rho = \{1\}$, this improves the $3^{\text{tw}} \cdot n^{O(1)}$ algorithm of van Rooij to $2^{\text{tw}} \cdot n^{O(1)}$.

Despite the unusually delicate definition of $c_{\sigma, \rho}$, we show that our algorithms are most likely optimal, that is, for any pair (σ, ρ) of finite or cofinite sets where the problem is non-trivial, and any $\varepsilon > 0$, a $(c_{\sigma, \rho} - \varepsilon)^{\text{tw}} \cdot n^{O(1)}$ -algorithm counting the number of (σ, ρ) -sets would violate the Counting Strong Exponential-Time Hypothesis (#SETH). For finite sets σ and ρ , our lower bounds also extend to the decision version, showing that our algorithms are optimal in this setting as well. In contrast, for many cofinite sets, we show that further significant improvements for the decision and optimization versions are possible using the technique of representative sets.

4 Working groups

4.1 Counting Functions via Extension Oracles

Marco Bressan (University of Milan, IT), Konrad Anand (Queen Mary University of London, GB), and Holger Dell (Goethe-Universität – Frankfurt am Main, DE)

License  Creative Commons BY 4.0 International license
© Marco Bressan, Konrad Anand, and Holger Dell

Introduction. Let $X = \{1, \dots, n\}$, and let \mathcal{C} be a set of functions from X to some set Y ; for the sake of this introduction we may assume $Y = \{0, 1\}$. A *partial function* from X to Y is a function $\hat{f} : X \rightarrow Y \cup \{\star\}$ where \star is a special symbol meaning abstention. An *extension oracle* or *consistency oracle* $O_{\mathcal{C}}$ for \mathcal{C} takes in input a partial function \hat{f} from X to Y and returns 1 if and only if that function has an extension in \mathcal{C} , i.e., if $\exists f \in \mathcal{C}$ such that $f(x) = \hat{f}(x)$ for all $x \in \hat{f}^{-1}(Y)$. We consider the following counting problem: given X, Y and access to $O_{\mathcal{C}}$, compute $|\mathcal{C}|$ or a good approximation to it. This problem arises for instance in machine learning, where \mathcal{C} is the concept class, or the version space (the set of concepts consistent with what the learner has seen so far). In this case, an efficient algorithm to compute or estimate $|\mathcal{C}|$ yields efficient algorithms for, say, the Halving algorithm [2, 1, 5].

Outcomes. First, we have highlighted a separation between the consistency oracle and the *independence oracle* for hypergraphs. The independence oracle was recently used by Dell, Lapinskas and Meek [4, 3] in the problem of counting the hyperedges of a hypergraph $H = (V, E)$. Clearly, this problem can be cast in our setting by letting $X = \{1, \dots, |V|\}$, $Y = \{0, 1\}$, and $\mathcal{C} = \{\mathcal{K}_e : e \in E\}$. An *independence oracle* for H takes in input a subset $S \subseteq V$ and returns 1 if and only if there exists $e \in E$ with $e \subseteq S$. In their work, the authors consider k -hypergraphs (ones where every edge has size k), and they show that even in that case one may need $|V|^{\Omega(k)}$ queries to approximate $|E|$ [4]. In fact it is easy to see that an independence oracle is not sufficient to solve the problem at all if the hypergraph is arbitrary: for instance, an independence oracle will output 1 on every input as long as E contains all singletons, and therefore any two such hypergraphs are indistinguishable via independence oracles. Instead, a consistency oracle always allows one to compute \mathcal{C} using $O(|\mathcal{C}|)$ queries, via a simple exhaustive search tree exploration. Thus in particular one gets a polynomial-time algorithm whenever $|\mathcal{C}| = n^{O(1)}$.

Second, we have obtained a construction that proves what follows. For every fixed $k \geq 1$, in order to be able to distinguish with non-vanishing probability between $|\mathcal{C}| = n^k$ and $|\mathcal{C}| = \Omega(n^{2k})$, one needs to make a number of calls to the consistency oracle of order:

$$\left(\frac{n}{2k \log n}\right)^{k+1} \quad (1)$$

This construction will likely be the basis for future developments.

References

- 1 Dana Angluin. Queries and concept learning. *Machine Learning*, 2(4):319–342, 1988. doi:10.1023/A:1022821128753.
- 2 J. M. Barzdin and R. V. Freivald. On the prediction of general recursive functions. *Soviet Math. Doklady*, 13:1224–1228, 1972.
- 3 Holger Dell, John Lapinskas, and Kitty Meeks. Approximately counting and sampling small witnesses using a colorful decision oracle. *SIAM Journal on Computing*, 51(4):849–899, 2022. arXiv:https://doi.org/10.1137/19M130604X, doi:10.1137/19M130604X.
- 4 Holger Dell, John Lapinskas, and Kitty Meeks. Nearly optimal independence oracle algorithms for edge estimation in hypergraphs, 2022. URL: https://arxiv.org/abs/2211.03874, doi:10.48550/ARXIV.2211.03874.
- 5 Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2(4):285–318, 1988. doi:10.1023/A:1022869011914.

4.2 Independent sets of fixed size

Ewan Davies (University of Colorado – Boulder, US), Sarah Cannon (Claremont McKenna College, US), Charlie Carlson (University of Colorado – Boulder, US), Sarah Miracle (University of St. Thomas – St. Paul, US), and Noela Müller (TU Eindhoven, NL)

License © Creative Commons BY 4.0 International license

© Ewan Davies, Sarah Cannon, Charlie Carlson, Sarah Miracle, and Noela Müller

Main reference Ewan Davies, Will Perkins: “Approximately Counting Independent Sets of a Given Size in Bounded-Degree Graphs”, in Proc. of the 48th International Colloquium on Automata, Languages, and Programming, ICALP 2021, July 12-16, 2021, Glasgow, Scotland (Virtual Conference), LIPIcs, Vol. 198, pp. 62:1–62:18, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

URL https://doi.org/10.4230/LIPICs.ICALP.2021.62

Recently, a number of algorithmic methods for approximately counting the number of independent sets of a given size $k = \lfloor \alpha n \rfloor$ in n -vertex bounded-degree graphs have been discovered. In [2] the sharp hardness threshold in the density α was uncovered, and the algorithm in the tractable region of densities is based on rejection sampling from the hard-core model. Alternative approaches based local central limit theorems that yield faster algorithms were given in [3].

There is a natural Markov chain whose stationary distribution is uniform on independent sets of size k , namely the *down-up walk* that from a state I takes a uniform random element $v \in I$ and moves to a uniform random independent set of size k containing $I \setminus \{v\}$. This was shown using path coupling [1] to mix rapidly at densities up to roughly $\alpha \sim 1/(2d)$ in graphs of maximum degree d . The hardness threshold from [2] is slightly larger: $\alpha_c \sim e/(1+e) \cdot 1/d$.

This working group focused on the question of whether the down-up walk mixes rapidly up to the hardness threshold. Emerging techniques that seem pertinent include spectral independence and localization schemes, but there is a rather significant obstacle to apply these techniques associated with the global constraint of a fixed size k . We did not overcome

the main obstacle, and instead explored alternative approaches and familiarised ourselves with topics such as correlation decay, computation trees, and zeros of partition functions for the fixed-size model. One starting point is an idea due to Heng Guo that provides a computation tree for the ratios $r_k(v) = \Pr_k(v \in I) / \Pr_k(v \notin I)$ where \Pr_k is over the uniform independent set of size k , but it remains unclear how to use this insight to study the mixing time of the down-up walk. We thank Guus Regts for an interesting discussion on techniques for finding zero-free regions.

References

- 1 R. Bubley and M. Dyer, *Path coupling: A technique for proving rapid mixing in Markov chains*, Proceedings 38th Annual Symposium on Foundations of Computer Science (Miami Beach, FL, USA), IEEE Comput. Soc, 1997, pp. 223–231.
- 2 E. Davies and W. Perkins, *Approximately Counting Independent Sets of a Given Size in Bounded-Degree Graphs*, 48th International Colloquium on Automata, Languages, and Programming (ICALP 2021) (Dagstuhl, Germany) (Nikhil Bansal, Emanuela Merelli, and James Worrell, eds.), Leibniz International Proceedings in Informatics (LIPIcs), vol. 198, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021, pp. 62:1–62:18.
- 3 V. Jain, W. Perkins, A. Sah, and M. Sawhney, *Approximate counting and sampling via local central limit theorems*, Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing (New York, NY, USA), STOC 2022, Association for Computing Machinery, June 2022, pp. 1473–1486.

4.3 Fine grained complexity of counting independent sets in bounded degree graphs


Heng Guo (University of Edinburgh, GB)

License  Creative Commons BY 4.0 International license
© Heng Guo

A key component of the approximate counting algorithm for independent sets in bounded degree graphs by Patel-Regts is a fixed parameter tractable algorithm for exactly counting them with the size being the parameter. The natural question is then if there is a corresponding lower bound. There are various hardness results when the degree bound goes to n , namely for general graphs. We have worked on reducing from those cases. However, there is a main difficulty, in that for general graphs, the size of the maximum independent set can be arbitrary, and yet for bounded degree graphs it is linear in the number of vertices. This makes such reductions difficult to construct.

4.4 Approximating the number of proper colorings of a planar graph with a large number of colors

Viresh Patel (Queen Mary University of London, GB), Andrei A. Bulatov (Simon Fraser University – Burnaby, CA), Charlie Carlson (University of Colorado – Boulder, US), Heng Guo (University of Edinburgh, GB), and Guus Regts (University of Amsterdam, NL)

License  Creative Commons BY 4.0 International license
 © Viresh Patel, Andrei A. Bulatov, Charlie Carlson, Heng Guo, and Guus Regts

The computational complexity of approximately counting the number of proper colorings with say a million colors of planar graphs is unclear. On the one hand as soon as we replace the number a million by any non-real number close arbitrarily to it and look at the evaluation of the chromatic polynomial at this point, this problem becomes #P-hard on planar graphs, as was recently proved in [1]. On the other hand a planar graph can be colored with 4 colors, which could possibly lead to the suspicion that for some large enough number of colors approximately counting the number of proper colorings should not be computational hard.


We have met in various compositions throughout the week and discussed the problem. Unfortunately we have not really made any progress. Some of the things we looked at include: the use of standard decomposition techniques for planar graphs and reductions to partition function of the ferromagnetic Potts model. The main conclusion from our initial discussions has been that this appears to be a difficult problem for which the current techniques are not powerful enough to make progress. Perhaps a useful question is to see if there exist sub exponential algorithms. For example Nederlof [2] designed an exact $2^{O(\sqrt{k})}\text{poly}(n)$ algorithm for counting the number of independent sets of size k in a planar n -vertex graph.

References

- 1 Ferenc Bencs, Jeroen Huijben, Guus Regts, Approximating the chromatic polynomial is as hard as computing it exactly, ArXiv preprint, <https://arxiv.org/pdf/2211.13790.pdf>, 2022.
- 2 Jesper Nederlof, Detecting and counting small patterns in planar graphs in subexponential parameterized time, *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing* (2020) 1293–1306.

4.5 Understanding the Homomorphism Basis

Marc Roth (University of Oxford, GB), Marco Bressan (University of Milan, IT), Radu Curticapean (IT University of Copenhagen, DK), Holger Dell (Goethe-Universität – Frankfurt am Main, DE), Jacob Focke (CISPA – Saarbrücken, DE), and Philip Wellnitz (MPI für Informatik – Saarbrücken, DE)

License  Creative Commons BY 4.0 International license
 © Marc Roth, Marco Bressan, Radu Curticapean, Holger Dell, Jacob Focke, and Philip Wellnitz

The goal of this working group was to improve the understanding of the structure of the so-called homomorphism basis of counting problems: Well-known transformations dating back to early works of Lovász allow the expression of a wide range of counting problems (including problems arising in database theory and network sciences) as a finite linear combination of homomorphism counts. For example, it is known that for every graph H there exists

a finitely supported function a from graphs to rationals such that, for every graph G the following is true:

$$\#\text{Sub}(H \rightarrow G) = \sum_F a(F) \cdot \#\text{Hom}(F \rightarrow G), \quad (2)$$

where $\#\text{Sub}(H \rightarrow G)$ denotes the number of subgraphs of G that are isomorphic to H , and $\#\text{Hom}(F \rightarrow G)$ denotes the number of graph homomorphisms (edge-preserving mappings) from F to G . In other words, the problem of counting copies of H in a graph G can be cast as computing a finite linear combination of homomorphism counts.

In 2017, Curticapean, Dell and Marx [1] established a remarkable property, sometimes called “complexity monotonicity”, of linear combinations as in (2): They are *exactly* as hard to compute as their hardest term $\#\text{Hom}(F \rightarrow G)$ with a non-zero coefficient ($a(F) \neq 0$).¹ Since the complexity of computing individual homomorphism counts is reasonably well-understood due to a result of Dalmau and Jonsson [2], this discovery enabled a general strategy for studying the complexity of counting problems: Cast the problem as a linear combination of homomorphism counts and investigate which terms cancel out, i.e., for which graphs F the coefficient $a(F)$ becomes 0. In recent years, this strategy has seen significant success in classifying counting problems arising e.g. in database theory [3, 4], subgraph and induced subgraph counting [1, 5, 6], modular counting [7, 8], fine-grained homomorphism counting [12], and pattern counting in degenerate graphs [9, 10, 11].

The purpose of this working group was to gather and unify the many different tools that have been established for analysing the homomorphism basis, to enhance the methods to tackle new problems, and to find common grounds for future collaborations.

Outcomes

- We discovered a new strategy for analysing the homomorphism basis of induced subgraph counting problems that enabled us to understand the complexity of various instances of the induced subgraph counting problem that have previously been unclassified. We hope that this strategy will enable us in the future to completely resolve the complexity of the induced subgraph counting problem as studied in [5, 6].
- We discovered that the general framework extends to pattern counting problems in hypergraphs (which is not surprising). However, we also found that we need new tools to understand the coefficients in the homomorphism basis, since the established tools do not always generalise to hypergraphs (which is surprising).

References

- 1 Radu Curticapean, Holger Dell, and Dániel Marx. Homomorphisms are a good basis for counting small subgraphs. In Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017, pages 210–223, 2017. doi:10.1145/3055399.3055502 .
- 2 Víctor Dalmau and Peter Jonsson. The complexity of counting homomorphisms seen from the other side. *Theor. Comput. Sci.*, 329(1-3):315–323, 2004. doi:10.1016/j.tcs.2004.08.008.
- 3 Hubie Chen and Stefan Mengel. Counting Answers to Existential Positive Queries: A Complexity Classification. In Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI

¹ This property is very special to homomorphism counts, as it does not hold for computing linear combinations of general counting problems: For example, computing the number of satisfying assignments plus the number of unsatisfying assignments of an n -variable CNF is always 2^n , i.e., easy to compute, although computing the individual terms is #P-hard.

- Symposium on Principles of Database Systems, PODS 2016, San Francisco, CA, USA, June 26 – July 01, 2016, pages 315–326, 2016. doi:10.1145/2902251.2902279 .
- 4 Holger Dell, Marc Roth, and Philip Wellnitz. Counting Answers to Existential Questions. In 46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece., pages 113:1–113:15, 2019. doi:10.4230/LIPIcs.ICALP.2019.113.
 - 5 Marc Roth, Johannes Schmitt, and Philip Wellnitz. Detecting and Counting Small Subgraphs, and Evaluating a Parameterized Tutte Polynomial: Lower Bounds via Toroidal Grids and Cayley Graph Expanders. In 48th International Colloquium on Automata, Languages, and Programming, ICALP 2021, July 12-16, 2021, Glasgow, Scotland (Virtual Conference), volume 198 of LIPIcs, pages 108:1–108:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.ICALP.2021.108.
 - 6 Jacob Focke and Marc Roth. Counting small induced subgraphs with hereditary properties. In Stefano Leonardi and Anupam Gupta, editors, STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 – 24, 2022, pages 1543–1551. ACM, 2022. doi:10.1145/3519935.3520008.
 - 7 J. A. Gregor Lagodzinski, Andreas Göbel, Katrin Casel, and Tobias Friedrich. On counting (quantum-)graph homomorphisms in finite fields of prime order. In Nikhil Bansal, Emanuela Merelli, and James 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 91:1–91:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.ICALP.2021.91 .
 - 8 Norbert Peyerimhoff, Marc Roth, Johannes Schmitt, Jakob Stix, and Alina Vdovina. Parameterized (modular) counting and cayley graph expanders. In Filippo Bonchi and Simon J. Puglisi, editors, 46th International Symposium on Mathematical Foundations of Computer Science, MFCS 2021, August 23-27, 2021, Tallinn, Estonia, volume 202 of LIPIcs, pages 84:1–84:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.MFCS.2021.84.
 - 9 Suman K. Bera, Lior Gishboliner, Yevgeny Levanzov, C. Seshadhri, and Asaf Shapira. Counting subgraphs in degenerate graphs. *J. ACM*, 69(3):23:1–23:21, 2022. doi:10.1145/3520240.
 - 10 Marco Bressan and Marc Roth. Exact and approximate pattern counting in degenerate graphs: New algorithms, hardness results, and complexity dichotomies. In 62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022, pages 276–285. IEEE, 2021. doi:10.1109/FOCS52979.2021.00036.
 - 11 Marco Bressan, Matthias Lanzinger, and Marc Roth. The complexity of pattern counting in directed graphs, parameterised by the outdegree. CoRR, abs/2211.01905, 2022. arXiv:2211.01905 , doi:10.48550/arXiv.2211.01905.
 - 12 Hubie Chen, Radu Curticapean, and Holger Dell. The exponential-time complexity of counting (quantum) graph homomorphisms. In Ignasi Sau and Dimitrios M. Thilikos, editors, Graph-Theoretic Concepts in Computer Science – 45th International Workshop, WG 2019, Vall de Núria, Spain, June 19-21, 2019, Revised Papers, volume 11789 of Lecture Notes in Computer Science, pages 364–378. Springer, 2019. doi:10.1007/978-3-030-30786-8_28.

Participants

- Konrad Anand
Queen Mary University of
London, GB
- Nima Anari
Stanford University, US
- Miriam Backens
University of Birmingham, GB
- Andreas Björklund
Lund, SE
- Marco Bressan
University of Milan, IT
- Andrei A. Bulatov
Simon Fraser University –
Burnaby, CA
- Sarah Cannon
Claremont McKenna College, US
- Charlie Carlson
University of Colorado –
Boulder, US
- Amin Coja-Oghlan
TU Dortmund, DE
- Radu Curticapean
IT University of
Copenhagen, DK
- Ewan Davies
University of Colorado –
Boulder, US
- Holger Dell
Goethe-Universität –
Frankfurt am Main, DE
- Martin Dyer
University of Leeds, GB
- Jacob Focke
CISPA – Saarbrücken, DE
- Andreas Galanis
University of Oxford, GB
- Andreas Göbel
Hasso-Plattner-Institut,
Universität Potsdam, DE
- Leslie Ann Goldberg
University of Oxford, GB
- Heng Guo
University of Edinburgh, GB
- Mark R. Jerrum
Queen Mary University of
London, GB
- Petteri Kaski
Aalto University, FI
- John Lapinskas
University of Bristol, GB
- Sarah Miracle
University of St. Thomas –
St. Paul, US
- Haiko Müller
University of Leeds, GB
- Noela Müller
TU Eindhoven, NL
- Marcus Pappik
Hasso-Plattner-Institut,
Universität Potsdam, DE
- Viresh Patel
Queen Mary University of
London, GB
- Guus Regts
University of Amsterdam, NL
- Marc Roth
University of Oxford, GB
- Philip Wellnitz
MPI für Informatik –
Saarbrücken, DE
- Stanislav Živný
University of Oxford, GB

