COMPUTATIONAL ASPECTS OF LARGE-SCALE PROBLEMS IN DISCRETE MATHEMATICS

SEPTEMBER 25 – SEPTEMBER 29
2026

BOOKLET OF ABSTRACTS

Program

Saturday morning, September 26

How can computers help in obtaining new results in graph theory?

(Invited Lecture)

Jorik Jooken

Leiden University, the Netherlands & KU Leuven, Belgium

This talk consists of two parts. In the first part, I will give an introduction to the domain of computer-assisted graph theory, which is concerned with developing algorithms in order to help researchers gain insights into various graph theoretical questions. In particular, I will discuss a broad range of popular techniques from this domain and briefly talk about their applications. In the second part, I will zoom in on an algorithm for exhaustively generating graphs and show how executing this algorithm leads to new insights in the domain of graph colouring.

This talk is based on two papers, available at arXiv:2508.20825 and arXiv:2404.11704.

About the speaker. Jorik Jooken is an FWO Postdoctoral Fellow at KU Leuven, working in the research group of Jan Goedgebeur. He earned his Ph.D. from KU Leuven in 2023, where he has been a postdoctoral researcher since his graduation. As of August 2026, he will join Leiden University in the Netherlands as an Assistant Professor.

Apart from his activities at KU Leuven, he has also spent several months in various universities abroad such as Comenius University (Slovakia), Nankai University (China), Western Sydney University (Australia) and Durham University (United Kingdom).

He is an expert in the field of discrete algorithms for combinatorial problems, with a particular emphasis on computer-assisted graph theory. He collaborates frequently with other mathematicians and computer scientists on various topics such as combinatorial optimization, extremal graph theory, graph coloring, long cycles in graphs, and enumeration of graphs. The topics of some of his latest publications include vertex-critical (P_5, W_4)-free graphs, (k, g)-graphs without $(g+1)$-cycles, girth and connectivity of cubic graphs with a unique longest cycle and a survey on computer-assisted graph theory.

He has been the advisor to three doctoral students and to numerous bachelor's and master's students.

New computational results for the degree/diameter problem for 4-valent Cayley graphs

Martin Mačaj

Comenius University in Bratislava

The degree/diameter problem for Cayley graphs is the problem of finding, for any pair $\Delta \geq 3$ and $D \geq 2$ of integers, the largest possible order $N_{Cay}(\Delta, D)$ of a Cayley graph of degree Δ and diameter D. We present results of our exhaustive computer-aided search for 4-valent Cayley graphs constructed from groups in the Small Groups Library in GAP. We show that the existing lower bounds on $N_{Cay}(4, D)$ are tight for $D = 5, 6$ and present new lower bounds for $D = 7, 8$. We also present a new lower bound for $N_{Cay}(4, 12)$.

Why is the computation of quasi-Cayley deficiency so hard?

Pavol Kollár

Comenius University in Bratislava

In graph theory and especially in algebraic graph theory, one is often interested in all possible permutations of vertices V of a graph $\Gamma = (V, E)$, which preserve the (non-)adjacency of these vertices. These permutations are called graph automorphisms and using the map composition operation, they form a group – the automorphism group $\mathrm{Aut}(\Gamma)$. If it so happens that $\mathrm{Aut}(\Gamma)$ acts transitively on the set V, we say that Γ is a vertex-transitive graph – and precisely these graphs are the objects examined in our work.

When we have a vertex-transitive graph Γ, one can consider the question of "how close is Γ to being a Cayley graph?" This question has been examined previously, and in more generality since then. Combining the approaches and methods of these earlier works gives us a way of measuring this aforementioned "closeness" by computing the quasi-Cayley deficiency of a given graph Γ.

One of the issues is that, even though we have lists of small vertex-transitive graphs, the computation itself is a genuinely hard problem and one that ultimately comes down to a problem in Integer Linear Programming (ILP). In our work, we present partial results obtained by our algorithms and share a new idea that stems from Representation Theory that has the potential to make said ILP problem easier.

Hamiltonicity of Cages and Record Graphs

Jakub Šmihuľa

Comenius University in Bratislava

A (k, g)-cage is a k-regular graph of girth g with the smallest possible number of vertices. Although cages are among the most studied objects in extremal graph theory, the question of whether they are Hamiltonian remains open, and the supporting experiments have never been published. In this work we study the Hamiltonicity of all currently known cages and of a large collection of record graphs—the smallest known k-regular graphs of girth g. We first show that classical sufficient conditions for Hamiltonicity (Dirac, Ore, Erdős–Hobbs and Jackson) cannot be satisfied by cages, because their order grows far beyond the corresponding degree bounds. We therefore reduce the Hamiltonian cycle problem to the Travelling Salesman Problem and search for Hamiltonian cycles with the exact Concorde solver and, for the largest instances, with the Lin–Kernighan heuristic running on the Devana high-performance cluster. Every tested cage was found to be Hamiltonian with the single exception of the Petersen graph (the $(3, 5)$-cage), and every tested record graph—including the difficult cases $\mathrm{rec}(3, 18)$ and $\mathrm{rec}(3, 20)$—was Hamiltonian as well. We additionally implement Sachs' construction of regular Hamiltonian graphs of given girth and compare its orders with those of cages. Our results support the hypothesis that all cages are Hamiltonian except for the Petersen graph. They further indicate that any non-Hamiltonian record graph would necessarily be accompanied by a Hamiltonian (k, g)-graph of equal or smaller order, making such graphs particularly valuable in the search for cages.

Representing finite groups as automorphism groups of hypergraphs

Tatiana B. Jajcayová

Comenius University, Bratislava, Slovakia

In our presentation, we will report on our on-going project, joint work with Robert Jajcay and Isabel Hubard, on the classification of finite groups that can be represented as regularly acting full automorphism groups of 3-uniform hypergraphs. Given a regular action of a finite group G on a set V, we investigate the question of the existence of a 3-uniform hypergraph $H = (V, B)$ on the set V whose full automorphism group $\mathrm{Aut}(H)$ is equal to the group G in its regular action.

Our approach contains both theoretical and computational aspects. In particular, we build on the computational evidence obtained by Dominika Závacká. The theoretical approach makes use of the well-known Graphical Regular and Digraphical

Regular Representation Problems, and we are able to show the existence of the 3-uniform hypergraphs for all but a finite list of abelian and generalized dicyclic groups.

Saturday afternoon, September 26

Hybrid Exact Search for Optimum Clique Coverings

Petr Kovář and Yifan Zhang

VŠB – Technical University of Ostrava (Czech Republic)

We study optimum clique coverings of complete graphs using a combination of exact-search methods. The computation is split into three parts: finding decompositions, excluding smaller block counts, and checking the solver output. Dancing links (DLX) and integer linear programming (ILP) are used to find exact coverings with prescribed block counts. Boolean satisfiability (SAT) is used only after arithmetic, symmetry, and local-structure reductions have produced smaller infeasibility instances. All claimed decompositions are checked again by direct edge counting. We obtain $C^{\xi}(18, \{3, 4\}, 2) = 33$ and $C^{\xi}(19, \{3, 4\}, 2) = 35$ for exact coverings by triangles and quadruples. We also use the same method to prove the lower bound $C^{\xi}(33, \{3, 4, 5\}, 2) \geq 59$. The computational data show that DLX and ILP can exchange roles as the faster construction method; this is why both are kept in the search. The K_{33} computation shows how SAT can be used after several layers of filtering.

Methods for Graphlets Frequency Computation in Networks

Martin Nehéz and Teodora Šimon

Slovak University of Technology in Bratislava

In the talk, we address various methods for counting the occurrence of graphlets in networks. In addition to algorithmic methods, we will focus on probabilistic methods. We estimate the counts of graphlets for sparse random Erdős–Rényi graphs.

Optimization of Search Spaces in the Cage Problem

Róbert Jajcay

Comenius University in Bratislava

Even though the Cage Problem is in its essence an optimization problem, due to the large orders of the graphs that need to be investigated, and the corresponding unmanageable number of graphs the girth of which needs to be determined, any kind of exhaustive search is completely out of question. As a result of this, researchers focus on a limited number of tried and successful methods they attempt to optimize with regard to the selection of their starting ingredients. This selection is nevertheless still based on a combination of ad hoc heuristics, trial and error, and a few mathematical insights often obtained only for a subclass of all possible inputs. In our talk, we will present some of these techniques, discuss the mathematics behind their use, and argue for a holistic approach taking into account the input ingredients as a whole and not one at a time.

Sunday, September 27

On the $((3n)_4)$-configurations of Kárteszi

György Kiss

ELTE, Budapest (Hungary) & University of Primorska, Koper (Slovenia)

An (n_k)-configuration $(\mathcal{P}, \mathcal{L})$ is a set $\mathcal{P}$ of n points and a set $\mathcal{L}$ of n lines such that each point lies on exactly k lines, and each line contains exactly k points. The configuration is geometric, topological, or combinatorial depending on whether lines are considered to be straight lines, pseudolines (simple closed curves) or just combinatorial lines. Configurations are closely related to finite geometries and extremal graph theory. In the first part of the talk, we review the known results related to the existence problem of configurations. Computational methods can help in solving the many open questions that arise here.

In 1964, *Ferenc Kárteszi* proved a theorem in real geometry that gives rise to a series of geometric 4-configurations $\mathrm{K}(n; \ell, m)$. In the second part of the talk, we consider the Kárteszi configurations and in particular show that $\mathrm{K}(7; 2, 3)$ is isomorphic to the famous (21_4)-configuration of *Grünbaum* and *Rigby*, which was only presented in 1990.

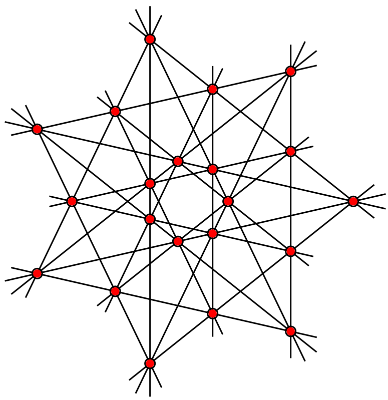

Figure 1: The Kárteszi configuration $\mathrm{K}(7; 2, 3)$

[1] G. Gévay, Gy. Kiss and T. Pisanski: *The Grünbaum-Rigby configuration as a special Kárteszi configuration,* arXiv:2512.18872.

On the order-diameter ratio of girth-diameter cages

Stijn Cambie, Jan Goedgebeur, Jorik Jooken, Tibo Van den Eede

KU Leuven, Belgium

For integers k, g, d, a $(k; g, d)$-cage (or simply girth-diameter cage) is a smallest k-regular graph of girth g and diameter d (if it exists). The order of a $(k; g, d)$-cage is denoted by $n(k; g, d)$. We determine asymptotic lower and upper bounds for the ratio between the order and the diameter of girth-diameter cages as the diameter goes to infinity. Moreover, we design and implement an exhaustive graph generation algorithm and use it to determine the exact order of several open cases and obtain – often exhaustive – sets of the corresponding girth-diameter cages. The largest case we generated and settled with our algorithm is a $(3; 7, 35)$-cage of order 136.

Full paper: arXiv:2511.21144

[1] S. Cambie, J. Goedgebeur, J. Jooken, and T. Van den Eede. On the Order-Diameter Ratio of Girth-Diameter Cages. In: Kozik, J., Wolff, A. (eds) SOFSEM 2026: Theory and Practice of Computer Science. SOFSEM 2026. Lecture Notes in Computer Science, vol 16448. Springer, Cham.

Properties of vertex-transitive graphs that are computationally easy to verify

Štefánia Glevitzká

Comenius University in Bratislava

In general, for a given graph, it is not computationally easy to determine whether it is vertex-transitive or not, since it usually requires determining a substantial part of the graph's automorphism group. However, vertex-transitive graphs need to satisfy numerous combinatorial properties, and those can be helpful in providing negative answers. For example, all vertex degrees must be identical, vertex neighbourhoods have to be isomorphic, each vertex needs to be contained in the same number of cycles of given length. Many of these properties have the advantage that they are easy to check using computer, since they address a property that can be checked locally, vertex by vertex.

In our talk, we focus on local properties related to cycles, in particular girth-regularity and conditions on the spectrum of cycles. We discuss how SageMath can be used to easily verify these properties and also how it can be helpful in the research process.

AutoGraphForge: Towards Automated Graph Theory Discovery

Ján Pastorek

Comenius University in Bratislava

We report on our ongoing project to develop a computational pipeline *AutoGraphForge* for an automated graph-theoretic conjecturing-refuting-formalizing-proving system. Generation of conjectures is counterexample-guided and runs in rounds: a Graffiti3 generator proposes conjectures over a small, evolving snapshot table T (initially a few hundred graphs with their computed invariants) that grows only by counterexamples to its own conjectures. A *novelty filter* of 559 classical and folklore relations, closed under transitive composition and linear identity substitution, decides via a linear program whether a candidate conjecture is already implied by known results. Then the conjecture is tested against a dataset of $\sim 348{,}000$ graphs unioning the complete House of Graphs invariant export, the exhaustive census of all connected graphs on at most nine vertices, and several extremal graph families (strongly regular, minimal Ramsey, Cayley, cages, barbells, lollipops, spiders, etc.), augmented by graphs generated by random models (random-regular, $G(n, p)$, random trees and random bipartite graphs). Then, *counterexample-search algorithms* attempt to find counterexamples to surviving conjectures. Run for several rounds on an HPC cluster, the loop yields $6{,}522$ surviving conjectures that no graph in the refutation dataset, which were not flagged as known by the novelty filter, and no active-search run eliminated—among them nontrivial relations between the annihilation number and the edge-cover number for bipartite graphs and regular graphs which we were able to prove by hand. A subsequent *formalization and proving* stage deterministically translates each surviving conjecture into a Lean 4 statement skeleton; every candidate proof is kernel-verified against a pinned `mathlib4` and our custom invariant preamble. The formalization-and-proving stage integrates two state-of-the-art neural provers—DeepSeek-Prover-V2-671B (served with vLLM) and the Lean-specialised OProver-32B— behind this independent kernel check. This stage is implemented end-to-end and passes initial sanity checks—the deterministic export yields type-correct Lean statements and the provers independently kernel-verify trivial inequalities—with the full pipeline currently running on the cluster.