Mathematisches Forschungsinstitut Oberwolfach

Report No. 15/2024

DOI: 10.4171/OWR/2024/15

Proof Complexity and Beyond

Organized by
Albert Atserias, Barcelona
Meena Mahajan, Chennai
Jakob Nordström, Copenhagen/Lund
Alexander Razborov, Chicago

24 March – 29 March 2024

Abstract. Proof complexity is a multi-disciplinary research area that addresses questions of the general form "how difficult is it to prove certain mathematical facts?" The current workshop focussed on recent advances in our understanding that the analysis of an appropriately tailored concept of "proof" underlies many of the arguments in algorithms, geometry or combinatorics research that make the core of modern theoretical computer science. These include the analysis of practical Boolean satisfiability (SAT) solving algorithms, the size of linear or semidefinite programming formulations of combinatorial optimization problems, the complexity of solving total **NP** search problems by local methods, and the complexity of describing winning strategies in two-player round-based games, to name just a few important examples.

Mathematics Subject Classification (2020): Primary: 03F20; Secondary: 68Q06, 68Q11, 68Q17.

Introduction by the Organizers

The workshop *Proof Complexity and Beyond* was organised by Albert Atserias (Barcelona), Jakob Nordström (Copenhagen/Lund), Meena Mahajan (Chennai) and Alexander Razborov (Chicago). The workshop was held during March 24th-29th and had 48 participants, including 4 remote participants. The program featured a total of 33 talks: 7 one-hour invited talks and 26 short talks. In addition, there was an open problem session, and during breaks intensive interaction took place in smaller groups.

As originally conceived by Stephen Cook and Robert Reckhow in their seminal article, propositional proof complexity is "the study of the length of the shortest proof of a propositional tautology in various proof systems as a function of the length of the tautology." The original motivation for what came to be known as Cook's program was to shed light on the celebrated **P** vs. **NP** problem, today one of the Clay Mathematical Institute Millenium Problems. But since that time, numerous connections with other areas—such as combinatorial optimization, practical Boolean satisfiability (SAT) solving, operation research, and extremal combinatorics, to name just a few—have been discovered. In our workshop, we attempted to "blend" Cook's program proper and these more modern directions.

We now proceed to describing concrete talks delivered at the workshop, and we attempt to loosely classify them into several categories. Our description is followed by abstracts that are listed in the order of appearance at the workshop.

Algebraic and semi-algebraic proof systems. Just as was the case with its predecessor (*Proof Complexity and Beyond*, 2017 edition), this was a major theme of our workshop, although the focus slightly shifted towards algebraic proof systems. It stems from the following simple observation. Two most fundamental mathematical results underlying algebraic and real geometry, Hilbert's Nullstellensatz and Stengle's Positivestellensatz, are essentially proof systems for proving unsatisfiability of a system of polynomial equations and inequalities, respectively. In turn, the grading of "proofs" in such proof systems by their "complexity" underlies several of the successful applications of these results of classical mathematics to theoretical computer science and affine areas.

The plenary talk by Risse reported a remarkable progress on the notoriously difficult "Small Clique Problem" in the context of semi-algebraic proof systems. Namely, the problem has been solved for the prominent system of Sherali-Adams (with an additional restriction on coefficients). Conneryd spoke about another recent major progress in the area, optimal polynomial calculus lower bounds for non-colorability of sparse random graphs. In a somewhat similar vein, Potechin reported an intriguing lower bound on the size of coefficients for the widely studied pigeon-hole-principle and the most basic algebraic proof system, Nullstellensatz.

Other talks in this category were more open-ended. Tzameret spoke about the functional lower bound method for analyzing strong proof systems and its applications. The talk by Toran was devoted to one of the most prominent tools (pebbling tautologies) in the context of algebraic proof systems; the method led to new strong separations between several such systems. The talk by Galesi highlighted important connections between complexity of algebraic proofs and the Tensor Isomorphism Problem, the latter being instrumental in several different areas. Hirsch introduced so-called "tropical proof systems" that provide a very nice connection between (semi)-algebraic proof complexity and tropical mathematics. Bonacina surveyed generalizations of the Max-SAT resolution proof system when weights can be negative and showed how it leads to new algebraic and semi-algebraic proof systems.

Proof, circuit, and communication complexity. Loosely speaking, circuit (or computational) complexity studies what we can compute with limited resources,

communication complexity studies how much information we must exchange to compute in settings where the input is distributed, and proof complexity studies what we can *prove* efficiently using only specific constructions that are allowed by circuit and/or communication constraints. Intuitively, these tasks should be related to each other. Quite remarkably, it turns out that these connections go way deeper than this loose philosophical speculation, and this made for another major theme at the workshop.

In her invited talk, DE REZENDE spoke about lifting techniques, a major development connecting these three areas in particularly influential ways. Her talk provided a game-theoretical perspective of the subject. Another invited talk given by SOKOLOV reviewed, in a somewhat similar style, applications of sunflowers in proof and circuit complexities. Initiated by Erdős and Rado in the context of extremal combinatorics, this notion has become indispensable in several areas of theoretical computer science. GÖÖS spoke of very interesting applications of the hardness condensation method, previously employed in circuit and proof complexity, to some prominent problems in the communication complexity setting. SOFRONOVA presented top-down lower bounds for depth 4 circuits that make an important step towards resolving one of the most fundamental (and notoriously difficult) problems in circuit complexity. HRUBEŠ spoke about a well-known system of monotone calculus and showed various connections to monotone arithmetic circuits.

Logic-based propositional proof systems. These are proof systems in the proper sense, i.e. those that encode normal mathematical arguments in a recognizable form. Most talks in this section can be viewed as steps towards Cook's program.

ITSYKSON spoke about their recent major result that had generated quite a bit of interest in the community: strong lower bounds for the regular variant of the system "parity resolution". PAPAMAKARIOS showed how to extend a (relatively) recent breakthrough result about non-automatability of resolution to bounded-depth Frege: this is now the strongest proof system for which this can be reasonably hoped to be achieved with our current understanding. HÅSTAD viewed the same system from a different angle: his talk reported lower bounds for the pigeonhole-principle on the grid. PANG considered the notion of very strong trade-offs previously established in circuit and proof complexity in several different contexts and showed how to extend it to a fundamental tool in graph isomorphism testing known as Weisfeiler-Leman algorithms.

Practical SAT Solving and Quantified Boolean Formulas (QBFs). The tremendous success of SAT solvers on real-world instances has thrown up newer challenges and opened new directions. In a survey talk aimed at highlighting one such challenge, NORDSTRÖM discussed the issue of certifying answers of SAT solvers and other combinatorial solvers in succinct yet easily verifiable formats. THAPEN-II presented evidence that an approach using symmetry-breaking techniques for SAT solving is plausibly stronger than extended resolution, since it can encapsulate limited reasoning with QBFs.

In a session looking at directions beyond propositional proofs, Mahajan gave a brief overview of proof complexity for the more succinct formalism of QBFs. Choudhury reported on initial steps towards analysing the dependency schemes heuristic, used in many QBF solvers, from a rigorous proof-theoretic viewpoint. Kache laid out the roadmap for obtaining lower bounds for the quantified version of polynomial calculus.

Proof Complexity and Search Problems. This is yet another fascinating connection which has seen a surge of advances and new insights after 2017. In his plenary talk opening the workshop, Robere surveyed this theory and known results, as well as outlined next steps to take. Fleming reported their recent contribution affirming that the prominent class of search problems PPP is not Turing-closed in the relevant (black-box) setting, thus providing strong evidence that its "big" version is also not Turing-closed. Thapen-I connected this theory to the feasible disjunction/feasible interpolation property that in particular led to the first example of a proof system for which these two concepts behave very differently.

Mathematical Logic, Meta-Complexity etc. In this category we list several talks that are particularly well connected to the parent discipline, mathematical logic. Oliveira surveyed the connections between (un)provability of statements in various theories and the complexity of proving computational lower bounds, presenting this study as a principled approach to understanding the difficulty of proving lower bounds. Along these lines, Carmosino discussed the difficulty of proving known circuit lower bounds in specific theories; provability of the size hierarchy would then imply other circuit lower bounds. Jeřábek discussed the theory of exponential integer parts, described some properties of these theories, and, towards obtaining a finite axiomatization, presented an associated 2-player game over the integers. Pudlák described how the additional strength of narrow implicit proof systems based on quantified propositional calculi, using resolution to certify the implictly described proofs, corresponds to a jump in the hierarchy.

Miscellaneous (Beyond Proof Complexity). There were several talks on topics not directly in the area of proof complexity per se, but with significant non-trivial connections. Kothari delivered an exposition on the Kikuchi matrix method, that has found much use recently in solving problems about extremal combinatorics, and highlighted where and how verifying unsatisfiablity plays a role. Tulsiani followed up with describing a general framework for obtaining decoding algorithm, based on the Sum-of-Squares hierarchy of semidefinite programs and proofs. Blekherman discussed (the undecidability of) certifying the unsatisfiability of graph homomorphism inequality problems, highlighting an interplay between extremal combinatorics and real algebraic geometry. Grohe described a lower bound for the k-dimensional Weisfeiler-Leman algorithms for graph isomorphism testing referred to earlier; such lower bounds imply lower bounds in algebraic proof systems for formulas that formalise the existence of isomorphisms.

Open Problems Session

On Tuesday evening at 20h an Open Problems Session was held in the main auditorium. The session was attended by almost all participants, and was run by Albert Atserias, one of the organizers of the workshop. On Monday morning the participants were polled in a show of hands to express interest in having such a session. In keeping with similar traditions in comparable workshop series, also with the precedent settled in the previous edition of the Proof Complexity and Beyond worskhop of 2017 (Oberwolfach 1733), the suggested format was to allocate 10 slots of 5 minutes each. The slots would be filled on a first-come first-served basis by volunteering presenters. The reception was enthusiastic. By Tuesday afternoon all 10 slots had been filled with names of presenters. In the actual session on Tuesday we had one extra last-minute improvised open problem presentation. The session ended within schedule at 21h.

What follows is a list of succinct descriptions of the open problems that were presented. We made an effort to summarize each open problem with a single-sentence interrogation. The interrogation is followed by some short clarifications concerning the definitions, or by some known relevant facts on the state of the art.

1. Mika Göös. Is there a TFNP class beyond FP that is easy in the random oracle model? One can see that PPP and PLS are hard in the random oracle model. For example, it is well-known that it takes $\Omega(2^{n/2})$ queries to have a good chance of finding a collision (let alone a preimage of 0^n) in a random function from $\{0,1\}^n$ to $\{0,1\}^n$. Thus, PPP is hard in the random oracle model.

2. Robert Robere. Is it possible to randomly black-box reduce 3-PIGEON to 2-PIGEON? Here 3-PIGEON is the TFNP search problem of finding three out of $2N+1$ pigeons that map to the same out of N holes, and 2-PIGEON is the standard pigeonhole principle search problem of finding two out of $N+1$ pigeons that map to the same out of N holes; i.e., the defining problem of PPP.

3. Noah Fleming. Are Stabbing Planes and Cutting Planes, as proof systems for integer linear programming, equivalent up to quasi-polynomial simulations? It is known that Stabbing Planes quasi-polynomially simulates Cutting Planes. Recently proved lower bounds for Stabbing Planes suggest that a converse simulation might be possible too.

4. Greg Bleckermann. Is non-negativity of binomials of graph densities decidable? A binomial of graph densities is a formal expression of the form $a_1 t(G_1) + a_2 t(G_2)$ where a_1, a_2 are integers and G_1, G_2 are finite graphs. The graph-density function of G, denoted by $t(G)$, is the graph function defined by $H \mapsto t(G,H) = |\mathrm{Hom}(G,H)|/|V(H)|^{|V(G)|}$; i.e., $t(G,H)$ is the fraction of maps from $V(G)$ to $V(H)$ that are homomorphisms from G to H. By standard tricks, the only interesting case is $a_1 = 1$ and $a_2 = -1$, i.e., deciding if $t(G_1) \geq t(G_2)$.

5. Marco Carmosino. Does the two-sorted bounded-arithmetic theory VPV prove the Non-Deterministic Time Hierarchy Theorem? It is known that the theory

VPV is able to prove many theorems of computational complexity theory, including the Deterministic Time Hierarchy Theorem in the form $\mathrm{DTIME}(n^{2c+1}) \not\subseteq \mathrm{DTIME}(n^c)$, for all $c \geq 1$.

6. Marc Vinyals. Can one efficiently reduce the number of bits in the coefficients of an n-variable linear threshold function from, say, n^{100} bits down to n^2 bits? It is known that $O(n \log n)$ bits suffice but no efficient algorithm is known to achieve even polynomial compression.

7. Shuo Pang. In the Cops-Robber game on the $k \times m$ cylinder grid, how many rounds can the Robber survive if the number of cops is $5k/2$? It is known that $k+1$ cops can catch the robber but they need $\Omega(m)$ rounds, while $3k$ cops can catch the robber in $O(\log m)$ rounds.

8. Jakob Nordström. In the configuration model of Resolution with a *baby* version of the redundancy-based rule added, how would one prove lower bounds? In the *baby* version of the new rule, two restrictions are imposed: (1) the added clause does not introduce any new variables, and (2) the implication test underlying the rule is replaced by the much weaker containment test. In this model, the standard pigeonhole principle formulas and the standard Tseitin formulas are easy.

9. Paul Beame. For PCR, can we get a better degree-automating algorithm than Groebner basis which does not immediately kill the negated variables? In PCR, variables come in pairs x and $\overline{x}$ with the axiom $x+\overline{x}-1=0$ to avoid exponentially many monomials when representing clauses. In such a situation, the first thing the Groebner basis algorithm would do is to eliminate all occurrences of $\overline{x}$ and replace them by $1-x$, or all occurrences of x and replace them by $1-\overline{x}$. This defeats the whole purpose of introducing the negated variables $\overline{x}$.

10. Albert Atserias. Can one find PC/SA/SOS degree lower bounds for graph isomorphism formulas $\mathrm{ISO}(G,H)$ that do not come from CFI graphs or their small variants? Here $\mathrm{ISO}(G,H)$ is a standard CNF encoding of the statement that G and H are isomorphic. A concrete question: Writing $\mathrm{vc}(G)$ for the minimum vertex cover size of G, find n-vertex graphs G_n and H_n with $\mathrm{vc}(G_n) < 0.51n$ and $\mathrm{vc}(H_n) > 0.99n$ yet the SA-degree of refuting $\mathrm{ISO}(G_n, H_n)$ grows unbounded. It is known, but not trivial, that such examples for SA would lift to similar examples for SOS.

11. Antonina Kolokolova. How strong is a deductive proof system in which lines are represented by DNNF (Decomposable Negation Normal Form) formulas? Such a system would simulate OBDD proofs, and also proofs with lines represented by read-once branching programs (aka free BDDs), even if they are not ordered.

Acknowledgement: The MFO and the workshop organizers would like to thank the National Science Foundation for supporting the participation of junior researchers in the workshop by the grant DMS-2230648, "US Junior Oberwolfach Fellows".

Workshop: Proof Complexity and Beyond

Table of Contents

Abstracts

Propositional Proof Complexity and TFNP

Robert Robere

A recent line of work [1–8] has demonstrated many deep connections between propositional proof systems and total NP search problems (TFNP). The basic correspondence allows us to associate a total search problem S with each propositional proof system P such that the following holds: for every tautology T, T has a short proof in P if and only if proving T can be "efficiently reduced" to proving the totality of S. This allows us to define a theory of reducibility for proof systems that is analogous to classical reducibility in complexity theory, it has led to the resolution of a number of open problems in both proof complexity and the theory of TFNP, and also has suggested new directions of study in both of these areas.

In this talk we will survey this connection, the recent progress that has been made, and outline some next steps for the development to take.

References

[1] Sam Buss, Noah Fleming, and Russell Impagliazzo. Tfnp characterizations of proof systems and monotone circuits. *Electron. Colloquium Comput. Complex.*, TR22-141, 2022.

[2] Samuel R. Buss and Alan S. Johnson. Propositional proofs and reductions between NP search problems. *Annals of Pure and Applied Logic*, 163(9):1163–1182, 2012.

[3] Ben Davis and Robert Robere. Colourful TFNP and propositional proofs. In Amnon Ta-Shma, editor, *38th Computational Complexity Conference, CCC 2023, July 17-20, 2023, Warwick, UK*, volume 264 of *LIPIcs*, pages 36:1–36:21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.

[4] Mika Göös, Alexandros Hollender, Siddhartha Jain, Gilbert Maystre, William Pires, Robert Robere, and Ran Tao. Further collapses in TFNP. In *Proceedings of the 37th Computational Complexity Conference (CCC)*, pages 33:1–33:15, 2022.

[5] Mika Göös, Alexandros Hollender, Siddhartha Jain, Gilbert Maystre, William Pires, Robert Robere, and Ran Tao. Separations in proof complexity and TFNP. *Electron. Colloquium Comput. Complex.*, TR22-058, 2022.

[6] Mika Göös, Pritish Kamath, Robert Robere, and Dmitry Sokolov. Adventures in monotone complexity and TFNP. In *Proceedings of the 10th Innovations in Theoretical Computer Science Conference (ITCS)*, volume 124, pages 38:1–38:19, 2018.

[7] Pavel Hubáček, Erfan Khaniki, and Neil Thapen. TFNP intersections through the lens of feasible disjunction. In Venkatesan Guruswami, editor, *15th Innovations in Theoretical Computer Science Conference, ITCS 2024, January 30 to February 2, 2024, Berkeley, CA, USA*, volume 287 of *LIPIcs*, pages 63:1–63:24. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024.

[8] Yuhao Li, William Pires, and Robert Robere. Intersection classes in TFNP and proof complexity. In Venkatesan Guruswami, editor, *15th Innovations in Theoretical Computer Science Conference, ITCS 2024, January 30 to February 2, 2024, Berkeley, CA, USA*, volume 287 of *LIPIcs*, pages 74:1–74:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024.

Black-Box PPP Is Not Turing Closed

Noah Fleming
(joint work with Stefan Grosser, Toniann Pitassi, Robert Robere)

The complexity class PPP contains all total search problems many-one reducible to the Pigeon problem, where we are given a succinct encoding of a function mapping $n+1$ pigeons to n holes, and must output two pigeons that collide in a hole. PPP is one of the "original five" syntactically-defined subclasses of TFNP [3,9–11], and has been extensively studied due to its defining problem — the pigeonhole principle — which captures strong induction, which is the basic axiom underlying most formal systems for mathematical reasoning. Additionally, PPP has strong connections to other areas such as the theory of lattices and cryptography [1, 7, 8, 13], extremal combinatorics [4, 12], and propositional proof complexity [2, 5, 6].

Despite the prominent role of PPP, it seems to lack certain robustness properties that all other natural TFNP classes enjoy. A prominent example of such a property is *closure under Turing reductions*. (A TFNP class $\mathcal{C}$ is closed under Turing reductions if any problem polynomial-time reducible to $\mathcal{C}$ via multiple calls to a problem in $\mathcal{C}$ is also polynomial-time reducible to a single call to $\mathcal{C}$.) The classical TFNP classes are typically defined using closure under *many-one* reductions, although, the original family of black-box separations between these classes, proved by [2], already hold for the Turing closed variants. The later work of Buss and Johnson [6] asked whether these classical TFNP classes are closed under Turing reductions. They proved that four of the five original TFNP classes PPA, PPAD, PPADS, and PLS are closed under Turing reductions, and they constructed an artificial TFNP subclass that was not Turing closed in the black-box setting. Subsequently, with the exception of PPP, all other natural TFNP classes have been shown to be Turing closed. Thus PPP stands as the only natural TFNP class not known to be Turing closed. The question of whether PPP contains its Turing closure was further highlighted by Daskalakis in his recent IMU Abacus Medal Lecture.

In this joint work with Stefan Grosser, Toniann Pitassi and Robert Robere, we prove that PPP is indeed not Turing-closed in the black-box setting, affirmatively resolving the above conjecture and providing strong evidence that PPP is not Turing-closed. In fact, we are able to separate PPP from its *non-adaptive* Turing closure, in which all calls to the Pigeon oracle must be made in parallel. This differentiates PPP from all other important TFNP subclasses, and especially from its closely-related subclass PWPP — defined by reducibility to the *weak* pigeonhole principle — which is known to be non-adaptively Turing-closed. Our proof requires developing new tools for PPP lower bounds, and creates new connections between PPP and the theory of *pseudoexpectation operators* used for Sherali-Adams and Sum-of-Squares lower bounds. In particular, we introduce a new type of pseudo-expectation operator that is precisely tailored for lower bounds against black-box PPP, which may be of independent interest.

References

[1] Frank Ban, Kamal Jain, Christos H. Papadimitriou, Christos-Alexandros Psomas, and Aviad Rubinstein. Reductions in PPP. *Inf. Process. Lett.*, 145:48–52, 2019.

[2] Paul Beame, Stephen Cook, Jeff Edmonds, Russell Impagliazzo, and Toniann Pitassi. The relative complexity of NP search problems. *Journal of Computer and System Sciences*, 57(1):3–19, 1998.

[3] Paul Beame, Russell Impagliazzo, Jan Krajíček, Toniann Pitassi, and Pavel Pudlák. Lower bounds on Hilbert's Nullstellensatz and propositional proofs. In *Proceedings of the 35th Symposium on Foundations of Computer Science (FOCS)*, pages 794–806, 1994.

[4] Romain Bourneuf, Lukáš Folwarczný, Pavel Hubáček, Alon Rosen, and Nikolaj I. Schwartzbach. Ppp-completeness and extremal combinatorics. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge, Massachusetts, USA*, volume 251 of *LIPIcs*, pages 22:1–22:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.

[5] Joshua Buresh-Oppenheim and Tsuyoshi Morioka. Relativized NP search problems and propositional proof systems. In *Proceedings of the 19th IEEE Conference on Computational Complexity (CCC)*, pages 54–67, 2004.

[6] Samuel R. Buss and Alan S. Johnson. Propositional proofs and reductions between NP search problems. *Annals of Pure and Applied Logic*, 163(9):1163–1182, 2012.

[7] Pavel Hubáček and Jan Václavek. On search complexity of discrete logarithm. 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 60:1–60:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[8] Emil Jeřábek. Integer factoring and modular square roots. *J. Comput. Syst. Sci.*, 82(2):380–394, 2016.

[9] David Johnson, Christos Papadimitriou, and Mihalis Yannakakis. How easy is local search? *Journal of Computer and System Sciences*, 37(1):79–100, 1988.

[10] Nimrod Megiddo and Christos Papadimitriou. On total functions, existence theorems and computational complexity. *Theoretical Computer Science*, 81(2):317–324, 1991.

[11] Christos Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. *Journal of Computer and System Sciences*, 48(3):498–532, 1994.

[12] Amol Pasarkar, Christos H. Papadimitriou, and Mihalis Yannakakis. Extremal combinatorics, iterated pigeonhole arguments and generalizations of PPP. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge, Massachusetts, USA*, volume 251 of *LIPIcs*, pages 88:1–88:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.

[13] Katerina Sotiraki, Manolis Zampetakis, and Giorgos Zirdelis. PPP-completeness with connections to cryptography. In *Proceedings of the 59th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 148–158, 2018.

TFNP Intersections and Feasible Disjunction

Neil Thapen

(joint work with Pavel Hubáček, Erfan Khaniki)

The complexity class CLS was introduced by Daskalakis and Papadimitriou [1] to capture the computational complexity of important TFNP problems solvable by local search over continuous domains and, thus, lying in both PLS and PPAD. It was later shown that, e.g., the problem of computing fixed points guaranteed by Banach's fixed point theorem is CLS-complete by Daskalakis et al. [2]. Recently,

Fearnley et al. [3] disproved the plausible conjecture of Daskalakis and Papadimitriou that CLS is a proper subclass of PLS ∩ PPAD by proving that CLS = PLS ∩ PPAD.

To study the possibility of other surprising collapses in TFNP, we connect classes formed as the intersection of existing subclasses of TFNP with the phenomenon of *feasible disjunction* in propositional proof complexity; where a proof system has the feasible disjunction property if, whenever a disjunction $F \vee G$ has a small proof, and F and G have no variables in common, then either F or G has a small proof [7, 8]. We study feasible disjunction for various systems and notions of smallness, in particular extending work of Hakoniemi [5] to show a kind of feasible disjunction for size and degree for Sherali Adams. Using this we separate the classes formed by intersecting the classical subclasses PLS, PPA, PPAD, PPADS, PPP and CLS, relying extensively on the lower bounds and connections with proof systems shown recently by Göös et al. [4]. We also give the first examples of proof systems which have the feasible interpolation property, but not the feasible disjunction property.

This work has appeared as [6].

References

[1] Constantinos Daskalakis and Christos H. Papadimitriou. *Continuous local search.* In Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, pp. 790–804, 2011.

[2] Constantinos Daskalakis, Christos Tzamos and Manolis Zampetakis. *A converse to Banach's fixed point theorem and its CLS-completeness.* In Proceedings of the ACM SIGACT Symposium on Theory of Computing, STOC 2018, pp. 44–50, 2018.

[3] John Fearnley, Paul Goldberg, Alexandros Hollender and Rahul Savani. *The complexity of gradient descent*: CLS = PPAD ∩ PLS. Journal of the ACM, 70(1):7:1–7:74, 2023.

[4] Mika Göös, Alexandros Hollender, Siddhartha Jain, Gilbert Maystre, William Pires, Robert Robere and Ran Tao. *Separations in proof complexity and TFNP.* In Proceedings of the IEEE Annual Symposium on Foundations of Computer Science, FOCS 2022, pp. 1150–1161, 2022.

[5] Tuomas Hakoniemi. Size bounds for algebraic and semialgebraic proof systems. PhD thesis, Universitat Politecnica de Catalunya, 2022.

[6] Pavel Hubáček, Erfan Khaniki and Neil Thapen. *TFNP Intersections Through the Lens of Feasible Disjunction.* In Innovations in Theoretical Computer Science Conference (ITCS 2024), LIPIcs Vol 287, pp. 63:1-63:24, 2024.

[7] Jan Krajíček. Bounded arithmetic, propositional logic, and complexity theory. Volume 60 of Encyclopedia of mathematics and its applications, Cambridge University Press, 1995.

[8] Pavel Pudlák. *On reducibility and symmetry of disjoint NP pairs.* Theoretical Computer Science, 295:323–339, 2003.

A Variant of Monotone Calculus

Pavel Hrubeš

Monotone calculus is a Frege-style system which operates with implications $A \to B$ where A and B are monotone. I will define a weakening of this system and show its connections with monotone arithmetic circuits.

References

[1] Albert Atserias, Nicola Galesi, and Pavel Pudlák. Monotone simulations of non-monotone proofs. *Journal of Computer and System Sciences*, 65:626–638, 2002.

[2] Pavel Hrubeš. On ϵ-sensitive monotone computations. *Computational Complexity*, 29(2), 2020.

Proof Complexity, Communication Complexity, and Lifting

Susanna F. de Rezende

Query-to-communication lifting theorems are methods of obtaining lower bounds for communication models from lower bounds for weaker query models. This method was used in [13] to separate the monotone NC hierarchy and to obtain monotone circuit depth lower bounds for functions like *st*-connectivity and *k*-clique. Soon after it was generalized in [2] to obtain lower bounds for tree-like proof systems. It was later brought to light in [6] where it was shown that this technique is more general than originally presented, and this gave rise to many new result in proof and circuit complexity, including: optimal $2^{\Omega(n)}$ lower bounds on the size of monotone boolean formulas computing an explicit monotone function in NP [10] and near optimal $2^{\widetilde{\Omega}(n)}$ for a function in monotone P [3]; the refinement of the monotone AC^i hierarchy from the monotone NC^i hierarchy and new tradeoffs for cutting planes proofs [4]; and a new family of techniques for proving lower bounds on cutting planes proofs and monotone circuit size [5].

In this talk we will view proofs and circuits as games: the *Prover-Adversary game* [1, 11] where a proof is seen as a protocol for solving the falsified clause search problem $\mathsf{Search}(F)$ for an unsatisfiable CNF formula F; and the *monotone Karchmer-Wigderson game* [7] where a monotone circuit is seen as communication (dag-like) protocol [8,9,12,14] to solve the monotone Karchmer-Wigderson relation $\mathsf{mKW}(f)$ for a monotone function f. We will see that, given an unsatisfiable CNF formula F, we can define a (partial) monotone function f such that $\mathsf{Search}(F)$ and $\mathsf{mKW}(f)$ become exactly the same problem, and then see how lifting theorems can be used to obtain lower bounds for protocols solving these search problems. In particular we will argue that for a formula F encoding the pigeonhole principle, we can obtain monotone circuit size and depth lower bounds for the clique-colouring function from communication lower bounds for $\mathsf{Search}(F)$ composed with the so-called indexing gadget. These communication lower bounds can, in turn, be shown via lifting theorems, such as those in [3,5,6,10,13], leading to the best known size and depth lower bounds for the clique-colouring function.

References

[1] Albert Atserias and Víctor Dalmau, *A combinatorial characterization of resolution width*, Journal of Computer and System Sciences **74** (2008), no. 3, 323–334.

[2] Maria Luisa Bonet, Juan Luis Esteban, Nicola Galesi, and Jan Johannsen, *On the relative complexity of resolution refinements and cutting planes proof systems*, SIAM Journal on Computing **30** (2000), no. 5, 1462–1484, Preliminary version in *FOCS '98*.

[3] Susanna F. de Rezende, Or Meir, Jakob Nordstrom, Toniann Pitassi, Robert Robere, and Marc Vinyals, *Lifting with simple gadgets and applications to circuit and proof complexity*, Proceedings of the 61st IEEE Annual Symposium on Foundations of Computer Science (FOCS '20), November 2020.
[4] Susanna F. de Rezende, Jakob Nordström, and Marc Vinyals, *How limited interaction hinders real communication (and what it means for proof and circuit complexity)*, Proceedings of the 57th IEEE Annual Symposium on Foundations of Computer Science (FOCS '16), October 2016.
[5] Ankit Garg, Mika Göös, Pritish Kamath, and Dmitry Sokolov, *Monotone circuit lower bounds from resolution*, Theory of Computing **16** (2020), no. 1, 1–30, Preliminary version in *STOC '18*.
[6] Mika Göös, Toniann Pitassi, and Thomas Watson, *Deterministic communication vs. partition number*, SIAM Journal on Computing **47** (2018), no. 6, 2435–2450, Preliminary version in *FOCS '15*.
[7] Mauricio Karchmer and Avi Wigderson, *Monotone circuits for connectivity require superlogarithmic depth*, SIAM Journal on Discrete Mathematics **3** (1990), no. 2, 255–265.
[8] Jan Krajíček, *Interpolation theorems, lower bounds for proof systems, and independence results for bounded arithmetic*, Journal of Symbolic Logic **62** (1997), no. 2, 457–486.
[9] Jan Krajíček, *Interpolation by a game*, Mathematical Logic Quarterly **44** (1998), no. 4, 450–458.
[10] Toniann Pitassi and Robert Robere, *Strongly exponential lower bounds for monotone computation*, Proceedings of the 49th Annual ACM Symposium on Theory of Computing (STOC '17), June 2017.
[11] Pavel Pudlák, *Proofs as games*, The American Mathematical Monthly **107** (2000), no. 6, 541–550.
[12] Alexander A. Razborov, *Unprovability of lower bounds on circuit size in certain fragments of bounded arithmetic*, Izvestiya: Mathematics (1995), 201–224.
[13] Ran Raz and Pierre McKenzie, *Separation of the monotone NC hierarchy*, Combinatorica **19** (1999), no. 3, 403–435, Preliminary version in *FOCS '97*.
[14] Dmitry Sokolov, *Dag-like communication and its applications*, Proceedings of the 12th International Computer Science Symposium in Russia (CSR '17), Lecture Notes in Computer Science, vol. 10304, Springer, June 2017, pp. 294–307.

On the Automatability of Bounded-Depth Frege Systems

Theodoros Papamakarios

A large chunk of research in proof complexity concentrates on trying to show that certain statements cannot have short proofs in some proof system. But even if a statement does have short proofs in a proof system, such proofs may not be easy to find. This motivates the notion of automatability [6]: A proof system P is called automatable if, given a statement τ, one can find a P-proof of τ in time polynomial in the size of the shortest P-proof of τ. Apart from being a natural notion in itself, of central importance to automated theorem proving, the concept of automatability is connected to other important threads in proof complexity, e.g. canonical pairs and feasible interpolation [2, 6, 11].

Now, the stronger the proof system, the harder it is to automate it, and indeed, early results show non-automatability for strong systems under plausible complexity theoretic assumptions [6, 10], and even, tightening the assumptions,

weaker systems [1,5]. More recently, starting with [3], weak proof systems, including resolution, res(k), cutting planes and various algebraic proof systems, have been shown to be as hard to automate as possible [3, 7–9]. In this talk, we argue how this can be extended to bounded-depth Frege systems.

We furthermore touch upon the problem of whether resolution is weakly automatable. A proof system P is weakly automatable if, given a statement τ, one can find a Q-proof of τ in time polynomial in the size of the shortest P-proof of τ, where Q is a proof system that polynomially simulates P. Whereas we know that resolution is as hard to automate as possible, whether it can be weakly automatable remains to a large extent open. An equivalent problem is whether depth-2 Frege systems have feasible interpolation [4, 5]. Focusing on the latter problem, we present a version of the point-line game of [4], present some of its properties, trying to suggest that the problem of weakly automating resolution might not be as hard as the problem of (strongly) automating resolution.

References

[1] Michael Alekhnovich and Alexander Razborov. Resolution is not automatizable unless W[P] is tractable. *SIAM Journal of Computing*, 38:1347–1363, 2008.

[2] Albert Atserias and Maria Luisa Bonet. On the automatizability of resolution and related propositional proof systems. *Information and Computation*, 189:182–201, 2004.

[3] Albert Atserias and Moritz Müller. Automating resolution is NP-hard. *Journal of the ACM*, 67:31:1–31:17, 2020.

[4] Arnold Beckmann, Pavel Pudlák, and Neil Thapen. Parity games and propositional proofs. *ACM Transactions on Computational Logic*, 15:17:1–17:30, 2014.

[5] Maria Luisa Bonet, Carlos Domingo, Ricard Gavaldà, Alexis Maciel, and Toniann Pitassi. Non-automatizability of bounded-depth frege proofs. *Computational Complexity*, 13:47–68, 2004.

[6] Maria Luisa Bonet, Toniann Pitassi, and Ran Raz. On interpolation and automatization for frege systems. *SIAM Journal of Computing*, 29:1939–1967, 2000.

[7] Susanna de Rezende, Mika Göös, Jakob Nordström, Toniann Pitassi, Robert Robere, and Dmitry Sokolov. Automating algebraic proof systems is NP-hard. In *Proccedings of the 53rd Annual ACM Symposium on Theory of Computing*, pages 209–222, 2021.

[8] Michal Garlík. Failure of feasible disjunction property for k-DNF resolution and NP-hardness of automating it. *Electronic Colloqium on Computational Complexity*, 2020.

[9] Mika Göös, Sajin Koroth, Ian Mertz, and Toniann Pitassi. Automating cutting planes is NP-hard. In *Proccedings of the 52nd Annual ACM Symposium on Theory of Computing*, pages 68–77, 2020.

[10] Jan Krajícek and Pavel Pudlák. Some consequences of cryptographical conjectures for S_2^1 and EF. *Information and Computation*, 140:82–94, 1998.

[11] Pavel Pudlák. On reducibility and symmetry of disjoint NP pairs. *Theoretical Computer Science*, 295:323–339, 2003.

Functional Lower Bounds in Algebraic Proofs: Symmetry, Lifting, and Barriers

Iddo Tzameret
(joint work with Tuomas Hakoniemi, Nutan Limaye)

Strong algebraic proof systems such as IPS (Ideal Proof System; Grochow–Pitassi [7]) offer a general model for deriving polynomials in an ideal and refuting unsatisfiable propositional formulas, subsuming most standard propositional proof systems. One of the most successful approach to this day for lower bounding the size of IPS refutations is the Functional Lower Bound Method (Forbes, Shpilka, Tzameret and Wigderson [5]), which reduces the hardness of refuting a polynomial equation $f(\bar{x}) = 0$ with no Boolean solutions to the hardness of computing the function $1/f(\bar{x})$ over the Boolean cube with an algebraic circuit. We consider this approach in general terms, and attempt to understand how far it can lead with respect to lower bounds, and where it cannot reach.

In particular, using symmetry we provide a general way to obtain many new hard instances against fragments of IPS via the functional lower bound method. This includes hardness over finite fields and hard instances different from Subset Sum variants both of which were unknown before, and stronger constant-depth lower bounds. Conversely, we expose the limitation of this method by showing it cannot lead to proof complexity lower bounds for any hard *Boolean* instance (e.g., CNFs) for any sufficiently strong proof systems. Specifically, we discuss the following new results:

Nullstellensatz degree lower bounds using symmetry: Extending [5] we show that every unsatisfiable symmetric polynomial with n variables requires degree $> n$ refutations (over sufficiently large characteristic). Using symmetry again, by characterising the $n/2$-homogeneous slice appearing in refutations, we show that unsatisfiable *invariant* polynomials of degree $n/2$ require degree $\geq n$ refutations.

Lifting to size lower bounds: Lifting our Nullstellensatz degree bounds to IPS-size lower bounds, we obtain exponential lower bounds for any polylogarithmic degree symmetric instance against IPS refutations written as oblivious read-once algebraic programs (roABP-IPS). For invariant polynomials, we show lower bounds against roABP-IPS and refutations written as multilinear formulas in the *placeholder* IPS regime (studied by Andrews-Forbes [2]), where the hard instances do not necessarily have small roABPs themselves, including over *positive characteristic* fields. This provides the first IPS-fragment lower bounds over finite fields.

By an adaptation of the work of Amireddy, Garg, Kayal, Saha and Thankey [1], we extend and strengthen the constant-depth IPS lower bounds obtained recently in Govindasamy, Hakoniemi and Tzameret [6] which held only for multilinear proofs, to $\mathsf{poly}(\log\log n)$ *individual degree* proofs. This is a natural and stronger constant depth proof system than

in [6], which admits small refutations for standard hard instances like the pigeonhole principle and Tseitin formulas.

Barriers for Boolean instances: While lower bounds against strong propositional proof systems were the original motivation for studying algebraic proof systems in the 1990s [3, 4], we show that the functional lower bound method alone cannot establish any size lower bound for *Boolean* instances for any sufficiently strong proof systems, and in particular, cannot lead to lower bounds against $\mathsf{AC}^0[p]$-Frege and TC^0-Frege.

Overall, this work wraps up to some extent research on IPS lower bounds via the functional lower bound method, showing how far it can be pushed, and where it cannot be applied. It generalises and improves previous work on IPS lower bounds obtained via the functional lower bound method in [5, 6]. We established size lower bounds for symmetric instances, and hard instances qualitatively different from previously known hard instances. This allows us also to show lower bounds over finite fields, which were open. We then showed how to incorporate recent developments on constant-depth algebraic circuit lower bounds [1] in the setting of proof complexity. This enables us to improve the constant-depth IPS lower bounds in [6] to stronger fragments, namely IPS refutations of constant depth and $\mathsf{poly}(\log \log n)$-individual degrees. As a corollary, we show a new finite field functional lower bound for *multilinear formulas* which may be of independent interest.

As for the barrier we uncovered, it is now evident that the functional lower bound method *alone* cannot be used to settle the long-standing open problems about the proof complexity of constant-depth propositional proofs with counting gates. This does not rule out however the ability of IPS lower bounds, and the IPS "paradigm" in general, to progress on these open problems, since other relevant methods may be found helpful (the meta-complexity method established in [9], the lower bounds for multiples method [2, 5], and the noncommutative reduction [8]). Moreover, our barrier only shows that we cannot hope to use a single non-Boolean unsatisfiable axiom $f(\bar{x}) = 0$ and consider the function $1/f(\bar{x})$ over the Boolean cube to obtain a CNF IPS lower bound (whenever the CNF is semantically implied from $f(\bar{x}) = 0$ over the Boolean cube). However, it does not rule out in general the use of a reduction to matrix rank, which is the backbone of many algebraic circuit lower bounds (as well as the functional lower bound method), and should potentially be helpful in proof complexity as well.

A very interesting problem that remains open is to prove CNF lower bounds using the functional method against fragments of IPS that sit below the reach of the barrier, namely fragments that cannot derive efficiently the conjunction of arbitrarily many polynomials (that is, systems that are not sufficiently strong in the above terminology).

References

[1] Prashanth Amireddy, Ankit Garg, Neeraj Kayal, Chandan Saha, and Bhargav Thankey. Low-depth arithmetic circuit lower bounds: Bypassing set-multilinearization. In Kousha

Etessami, Uriel Feige, and Gabriele Puppis, editors, *50th International Colloquium on Automata, Languages, and Programming, ICALP 2023, July 10-14, 2023, Paderborn, Germany*, volume 261 of *LIPIcs*, pages 12:1–12:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.

[2] Robert Andrews and Michael A. Forbes. Ideals, determinants, and straightening: Proving and using lower bounds for polynomial ideals. In *54th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2022*, 2022.

[3] Paul Beame, Russell Impagliazzo, Jan Krajíček, Toniann Pitassi, and Pavel Pudlák. Lower bounds on Hilbert's Nullstellensatz and propositional proofs. *Proc. London Math. Soc. (3)*, 73(1):1–26, 1996.

[4] Samuel R. Buss, Russell Impagliazzo, Jan Krajíček, Pavel Pudlák, Alexander A. Razborov, and Jiří Sgall. Proof complexity in algebraic systems and bounded depth Frege systems with modular counting. *Computational Complexity*, 6(3):256–298, 1996.

[5] Michael A. Forbes, Amir Shpilka, Iddo Tzameret, and Avi Wigderson. Proof complexity lower bounds from algebraic circuit complexity. *Theory Comput.*, 17:1–88, 2021.

[6] Nashlen Govindasamy, Tuomas Hakoniemi, and Iddo Tzameret. Simple hard instances for low-depth algebraic proofs. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, 2022.

[7] Joshua A. Grochow and Toniann Pitassi. Circuit complexity, proof complexity, and polynomial identity testing: The ideal proof system. *J. ACM*, 65(6):37:1–37:59, 2018.

[8] Fu Li, Iddo Tzameret, and Zhengyu Wang. Characterizing propositional proofs as noncommutative formulas. In *SIAM Journal on Computing*, volume 47, pages 1424–1462, 2018. Full Version: `http://arxiv.org/abs/1412.8746`.

[9] Rahul Santhanam and Iddo Tzameret. Iterated lower bound formulas: a diagonalization-based approach to proof complexity. In Samir Khuller and Virginia Vassilevska Williams, editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 234–247. ACM, 2021.

Clique Is Hard on Average for Sherali-Adams with Bounded Coefficients

Kilian Risse

(joint work with Susanna F. de Rezende, Aaron Potechin)

A fundamental problem of theoretical computer science is k-clique: given an n-vertex graph, determine whether it contains a clique of size k. This problem can be solved in time $O(n^k)$ by iterating over all subsets of vertices of size k and checking whether one of them is a clique. This naïve algorithm is essentially the fastest known; the constant in the exponent can be slightly improved [20] but, assuming the exponential time hypothesis [6], this linear dependence on k in the exponent is optimal in the worst-case.

Besides studying k-clique in the worst-case, one may consider it in the average-case setting. Suppose the given graph is an Erdős-Rényi graph with edge probability around the threshold of containing a k-clique. Does k-clique require time $n^{\Omega(k)}$ on such graphs? Or, even less ambitiously, is there an algorithm running in time $n^{o(k)}$ that decides the n^ϵ-clique problem on such graphs? It is unlikely that the hardness of such average-case questions can be based on worst-case hardness assumptions such as $\mathbf{P} \neq \mathbf{NP}$ or the exponential time hypothesis [5]. They are, in fact, being used as hardness assumptions themselves: the *planted clique conjecture*

states that $n^{1/2-\epsilon}$-clique requires time $n^{\Omega(\log n)}$ on Erdős-Rényi graphs with edge probability 1/2.

In order to obtain evidence that the planted clique conjecture holds we intend to prove it for bounded computational models. We consider the Sherali-Adams proof system with bounded coefficients and show that, for $k \leq 2 \log n$, it requires proofs of size $n^{\Omega(k)}$ to refute the claim that a uniformly sampled graph contains a clique of size $n^{0.1}$. This establishes a quantitative version of the planted clique conjecture for all algorithms captured by this proof system. Note that this proof system is incomparable to resolution, as shown in [12]. Previously similar results have been shown for tree-like resolution [4, 16] and for the Nullstellensatz proof system without dual-variables [18].

If we are only interested in refuting the existence of a smaller clique, say of size $4 \log n$, then there are essentially optimal $n^{\Omega(k)}$ average-case size lower bounds for regular resolution [1, 21]. For resolution, there are two average-case lower bounds that hold in different regimes: for $n^{5/6} \ll k \leq n/3$, Beame et al. [3] proved an average-case $\exp(n^{\Omega(1)})$ size lower bound and for $k \leq n^{1/3}$, Pang [21] proved a $2^{k^{1-o(1)}}$ lower bound. It is a long standing open problem, mentioned, e.g., in [4], to prove an unconditional $n^{\Omega(k)}$ resolution size lower bound for the unary encoding – even in the worst case.

For the less usual binary encoding of the clique formula it is somewhat straightforward to prove almost optimal $n^{\Omega(k)}$ resolution size lower bounds for the less usual binary encoding of the k-clique formula [17] and these lower bounds can even be extended to an $n^{\Omega(k)}$ lower bound for the Res(s) proof system for constant s [8].

An extended abstract previously appeared in the *Proceedings of the 64th Annual IEEE Symposium on Foundations of Computer Science (FOCS '23)* [10].

References

[1] Albert Atserias, Ilario Bonacina, Susanna F. de Rezende, Massimo Lauria, Jakob Nordström, and Alexander A. Razborov. Clique is hard on average for regular resolution. *J. ACM*, 68(4):23:1–23:26, 2021.

[2] Boaz Barak, Samuel Hopkins, Jonathan Kelner, Pravesh K. Kothari, Ankur Moitra, and Aaron Potechin. A nearly tight sum-of-squares lower bound for the planted clique problem. *SIAM Journal on Computing*, 48(2):687–735, 2019.

[3] Paul Beame, Russell Impagliazzo, and Ashish Sabharwal. The resolution complexity of independent sets and vertex covers in random graphs. *Comput. Complex.*, 16(3):245–297, 2007.

[4] Olaf Beyersdorff, Nicola Galesi, Massimo Lauria, and Alexander A. Razborov. Parameterized bounded-depth Frege is not optimal. *ACM Transactions on Computation Theory*, 4(3):7:1–7:16, September 2012. Preliminary version in *ICALP '11*.

[5] Andrej Bogdanov and Luca Trevisan. On worst-case to average-case reductions for NP problems. *SIAM J. Comput.*, 36(4):1119–1159, 2006.

[6] Jianer Chen, Xiuzhen Huang, Iyad A. Kanj, and Ge Xia. Linear FPT reductions and computational lower bounds. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC '04)*, pages 212–221, June 2004.

[7] Jianer Chen, Xiuzhen Huang, Iyad A. Kanj, and Ge Xia. Strong computational lower bounds via parameterized complexity. *J. Comput. Syst. Sci.*, 72(8):1346–1367, dec 2006.

[8] Stefan S. Dantchev, Nicola Galesi, Abdul Ghani, and Barnaby Martin. Proof complexity and the binary encoding of combinatorial principles. *CoRR*, abs/2008.02138, 2020.
[9] Stefan S. Dantchev and Barnaby Martin. Rank complexity gap for lovász-schrijver and sherali-adams proof systems. *Comput. Complex.*, 22(1):191–213, 2013.
[10] Susanna F. de Rezende, Aaron Potechin, and Kilian Risse. Clique is hard on average for unary sherali-adams. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 12–25, 2023.
[11] Rodney Downey and Michael R. Fellows. Fixed-parameter tractability and completeness II: Completeness for W[1]. *Theoretical Computer Science A*, 141(1–2):109–131, April 1995.
[12] Mika Göös, Alexandros Hollender, Siddhartha Jain, Gilbert Maystre, William Pires, Robert Robere, and Ran Tao. Separations in proof complexity and TFNP. In *63rd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2022, Denver, CO, USA, October 31 - November 3, 2022*, pages 1150–1161. IEEE, 2022.
[13] Dima Grigoriev, Edward A. Hirsch, and Dmitrii V. Pasechnik. Complexity of semi-algebraic proofs. In *STACS 2002*, pages 419–430, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.
[14] Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-SAT. *Journal of Computer and System Sciences*, 62(2):367–375, March 2001. Preliminary version in *CCC '99*.
[15] Pravesh K. Kothari, Ryuhei Mori, Ryan O'Donnell, and David Witmer. Sum of squares lower bounds for refuting any csp. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, page 132–145, New York, NY, USA, 2017. Association for Computing Machinery.
[16] Massimo Lauria. Cliques enumeration and tree-like resolution proofs. *Inf. Process. Lett.*, 135:62–67, 2018.
[17] Massimo Lauria, Pavel Pudlák, Vojtěch Rödl, and Neil Thapen. The complexity of proving that a graph is Ramsey. *Combinatorica*, 37(2):253–268, April 2017. Preliminary version in *ICALP '13*.
[18] Susan Margulies. *Computer Algebra, Combinatorics, and Complexity: Hilbert's Nullstellensatz and NP-complete Problems*. PhD thesis, University of California, Davis, 2008.
[19] Raghu Meka, Aaron Potechin, and Avi Wigderson. Sum-of-squares lower bounds for planted clique. In *Proceedings of the 47th Annual ACM Symposium on Theory of Computing (STOC '15)*, pages 87–96, June 2015.
[20] Jaroslav Nešetřil and Svatopluk Poljak. On the complexity of the subgraph problem. *Commentationes Mathematicae Universitatis Carolinae*, 026(2):415–419, 1985.
[21] Shuo Pang. Large clique is hard on average for resolution. In Rahul Santhanam and Daniil Musatov, editors, *Computer Science - Theory and Applications - 16th International Computer Science Symposium in Russia, CSR 2021, Sochi, Russia, June 28 - July 2, 2021, Proceedings*, volume 12730 of *Lecture Notes in Computer Science*, pages 361–380. Springer, 2021.
[22] Shuo Pang. SOS lower bound for exact planted clique. In Valentine Kabanets, editor, *36th Computational Complexity Conference, CCC 2021, July 20-23, 2021, Toronto, Ontario, Canada (Virtual Conference)*, volume 200 of *LIPIcs*, pages 26:1–26:63. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

Graph Colouring Is Hard on Average for Polynomial Calculus

Jonas Conneryd

(joint work with Susanna F. de Rezende, Jakob Nordström, Shuo Pang, Kilian Risse)

Determining the *chromatic number* of a graph G, i.e., how many colours are needed for the vertices of G if no two vertices connected by an edge should have the same colour, is one of the original 21 problems shown NP-complete in the seminal work

of Karp [12]. This *graph colouring problem*, as it is also referred to, has been extensively studied since then, but there are still major gaps in our understanding.

It is widely believed that any algorithm that colours graphs optimally has to run in exponential time in the worst case, and the currently fastest algorithm for 3-colouring has time complexity $O(1.3289^n)$ [4]. To understand graph colouring from the viewpoint of computational complexity, it is natural to investigate bounded models of computation that are strong enough to describe the reasoning performed by state-of-the-art algorithms for graph colouring and to prove unconditional lower bounds that hold in these models.

We investigate the hardness of graph colouring for algorithms based on algebraic reasoning, where the idea is to encode the graph colouring problem as a set of polynomials whose common roots correspond to proper colourings of the graph. The goal is then to either find those roots or prove that they do not exist. This leads us to the *polynomial calculus* proof system [1, 6], whose reasoning captures, for instance, most implementations of the Gröbner basis algorithm as well as an algorithm introduced in a well-known sequence of works [7–10] with surprisingly strong practical performance.

It was previously known [2, 13] that polynomial calculus requires linear degree, and hence exponential size via the size-degree relation [11], to solve graph colouring in the worst case. However, the hard instances in those papers come from reductions to other problems, so it was consistent with our knowledge that graph colouring is in fact easy for polynomial calculus except in some rather artificial special cases. Stronger evidence for the hardness of graph colouring would be an *average-case* lower bound, just as was established for resolution by Beame, Culberson, Mitchell, and Moore [3].

In this work we establish optimal, linear, degree lower bounds and exponential size lower bounds for polynomial calculus proofs of non-colourability of sparse random graphs. Our results hold over any field and for both Erdős-Rényi random graphs and random regular graphs.

An abridged version of this work appeared in the *Proceedings of the 64th Annual IEEE Symposium on Foundations of Computer Science (FOCS '23)* [5].

References

[1] Michael Alekhnovich, Eli Ben-Sasson, Alexander A. Razborov, and Avi Wigderson. Space complexity in propositional calculus. *SIAM Journal on Computing*, 31(4):1184–1211, April 2002. Preliminary version in *STOC '00*.

[2] Albert Atserias and Joanna Ochremiak. Proof complexity meets algebra. *ACM Transactions on Computational Logic*, 20:1:1–1:46, February 2019. Preliminary version in *ICALP '17*.

[3] Paul Beame, Joseph C. Culberson, David G. Mitchell, and Cristopher Moore. The resolution complexity of random graph k-colorability. *Discrete Applied Mathematics*, 153(1-3):25–47, December 2005.

[4] Richard Beigel and David Eppstein. 3-coloring in time $O(1.3289^n)$. *Journal of Algorithms*, 54(2):168–204, February 2005.

[5] Jonas Conneryd, Susanna F. de Rezende, Jakob Nordström, Shuo Pang, and Kilian Risse. Graph colouring is hard on average for polynomial calculus and Nullstellensatz. In *Proceedings of the 64th Annual IEEE Symposium on Foundations of Computer Science (FOCS '23)*, pages 1–11, November 2023.

[6] Matthew Clegg, Jeffery Edmonds, and Russell Impagliazzo. Using the Groebner basis algorithm to find proofs of unsatisfiability. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC '96)*, pages 174–183, May 1996.
[7] Jesús A. De Loera, Susan Margulies, Michael Pernpeintner, Eric Riedl, David Rolnick, Gwen Spencer, Despina Stasi, and Jon Swenson. Graph-coloring ideals: Nullstellensatz certificates, Gröbner bases for chordal graphs, and hardness of Gröbner bases. In *Proceedings of the 40th International Symposium on Symbolic and Algebraic Computation (ISSAC '15)*, pages 133–140, July 2015.
[8] Jesús A. De Loera, Jon Lee, Peter N. Malkin, and Susan Margulies. Hilbert's Nullstellensatz and an algorithm for proving combinatorial infeasibility. In *Proceedings of the 21st International Symposium on Symbolic and Algebraic Computation (ISSAC '08)*, pages 197–206, July 2008.
[9] Jesús A. De Loera, Jon Lee, Peter N. Malkin, and Susan Margulies. Computing infeasibility certificates for combinatorial problems through Hilbert's Nullstellensatz. *Journal of Symbolic Computation*, 46(11):1260–1283, November 2011.
[10] Jesús A. De Loera, Jon Lee, Susan Margulies, and Shmuel Onn. Expressing combinatorial problems by systems of polynomial equations and Hilbert's Nullstellensatz. *Combinatorics, Probability and Computing*, 18(4):551–582, July 2009.
[11] Russell Impagliazzo, Pavel Pudlák, and Jiří Sgall. Lower bounds for the polynomial calculus and the Gröbner basis algorithm. *Computational Complexity*, 8(2):127–144, 1999.
[12] Richard M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, The IBM Research Symposia Series, pages 85–103. Springer, 1972.
[13] Massimo Lauria and Jakob Nordström. Graph colouring is hard for algorithms based on Hilbert's Nullstellensatz and Gröbner bases. In *Proceedings of the 32nd Annual Computational Complexity Conference (CCC '17)*, volume 79 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 2:1–2:20, July 2017.

Pebble Games and Algebraic Proof Systems

Jacobo Torán
(joint work with Lisa-Marie Jaser)

Analyzing refutations of the well known pebbling formulas Peb(G) we prove some new strong connections between pebble games and algebraic proof system, showing that there is a parallelism between the reversible, black and black-white pebbling games on one side, and the three algebraic proof systems Nullstellensatz, Monomial Calculus [1] and Polynomial Calculus on the other side.

We prove that very similar results to those given in [3] for Nullstellensatz and reversible pebbling are also true for the case of Monomial Calculus and black pebbling. More concretely we show that for any DAG G with a single sink, if there is a MC refutation for Peb(G) having simultaneously degree s and size t then there is a black pebbling strategy on G with space s and time $t + s$. This is done by proving that any Horn formula has a very especial kind of MC refutation, which we call input monomial refutation since it is the same concept as an input refutation in Resolution.

For the other direction, we show that from a black pebbling strategy for G with space s and time t it is possible to extract a MC refutation for Peb(G) having simultaneously degree s and size ts. The small loss in the time parameter compared to the results in [3] comes from the fact that size complexity is measured

in slight different ways in NS and MC. Using these results we are able to show degree separations between NS and MC as well as strong degree-size tradeoffs for MC in the same spirit as those in [3]. The results also show that strong degree lower bounds for MC refutations do not imply exponential size lower bounds as it happens in the PC proof system [4].

The degrees of the refutation for pebbling formulas in NS and MS correspond exactly to the space in reversible and black games respectively. This is not the case for PC degree and space in the black-white pebble game [2]. We notice however that if instead of the degree we consider the complexity measure of variable space, then the connection still holds. For any single sink DAG G the variable space complexity of refuting Peb(G) in each of the algebraic proof systems NS, MC and PC is exactly the space needed in a strategy for pebbling G in each of the three versions reversible, black and black-white of the pebble game. This results allow us to apply known separations between the pebbling space needed in the different versions of the the game, in order to obtain separations in the variable space measure between the different proof systems.

References

[1] Christoph Berkholz and Martin Grohe. Limitations of algebraic approaches to graph isomorphism testing. In *ICALP 2015*, volume 9134 of *Lecture Notes in Computer Science*, pages 155–166. Springer, 2015.

[2] Joshua Buresh-Oppenheim, Matthew Clegg, Russell Impagliazzo, and Toniann Pitassi. Homogenization and the polynomial calculus. *Computational Complexity*, 11(3-4):91–108, 2002. Preliminary version in *ICALP '00*.

[3] Susanna F. de Rezende, Or Meir, Jakob Nordström, and Robert Robere. Nullstellensatz size-degree trade-offs from reversible pebbling. *Comput. Complex.*, 30(1):4, 2021.

[4] Russell Impagliazzo, Pavel Pudlák, and Jiří Sgall. Lower bounds for the polynomial calculus and the Gröbner basis algorithm. *Computational Complexity*, 8(2):127–144, 1999.

On Small-Depth Frege Proofs for PHP

Johan Håstad

We study formal proofs for the Pigeon Hole Principle (PHP). The PHP states that $m+1$ pigeons can fly to m holes such that no two pigeons fly to the same hole. It has $(m+1)m$ Boolean variables and variable x_{ij} is true iff pigeon i flies to the hole j. The axioms say that for each i there is a value of j such that x_{ij} is true and for each j there is at most one i such that x_{ij}. This is clearly a contradiction and the questions is whether this can be established by a short proof using simple and natural derivation rules and where each formula derived is of depth at most d in the basis given by $\wedge$ and $\vee$.

The case of $d = 1$ corresponds to resolution and an early milestone was obtained by Haken [3] in 1985 when he established exponential size lower bounds for such a proof. This was extended in a sequence of works [1,2,6,7] to prove that polynomial size proofs require depth d at least $\Omega(\log \log n)$.

These bounds remained the strongest lower bounds for any tautology until Pitassi et al [8] obtained super-polynomial lower bounds for depths up to $o(\sqrt{\log n})$.

The tautology considered was first studied by Tseitin [9] and considers a set linear equations modulo two defined by a graph. The underlying graph for [8] is an expander. These results were later extended to depth almost logarithmic by Håstad and Risse [4,5] and in this case the underlying graph is the two-dimensional grid.

We continue this line of research and prove that similar bounds apply to the PHP and to make use of previous work we study the graph PHP where the underlying graph is an odd size two-dimensional grid. If one colors this graph as a chess board, the corners are of the same color and let us assume this is white. In the graph PHP on the grid, there is a pigeon on each white square and it should fly to one of the adjacent black squares that define the holes.

Phrased slightly differently, the PHP on the grid says that there is a perfect matching of the odd size grid while Tseitin tautology on the same graph states that it is possible to assign Boolean values to the edges of the grid such that there is an odd number of true variables next to any node. As a perfect matching would immediately yield such an assignment, the PHP is a stronger statement and possibly easier to refute. The statements are, however, quite similar and indeed we our proof follow along the same lines as [5] and use many ideas from that paper.

As in most previous papers the main technical tool is to prove a "switching lemma". By assigning values to most variables in a formula it is possible to switch a small depth-two formula from being a CNF to being a DNF and the other way around. By choosing a very special way of assigning values we are able to preserve the graph PHP and hence prove our theorem by induction over d.

References

[1] Miklós Ajtai. The complexity of the pigeonhole principle. *Combinatorica*, 14(4):417–433, 1994. Preliminary version in *FOCS '88*.

[2] Stephen Bellantoni, Toniann Pitassi, and Alasdair Urquhart. Approximation and small-depth frege proofs. *SIAM J. Comput.*, 21:1161–1179, 1992.

[3] A. Haken. The intractability of resolution. *Theoretical Computer Science*, 39:297 – 308, 1985.

[4] J. Håstad. On small-depth frege proofs for tseitin for grids. *Journal of the ACM*, 68:1–31, 2020.

[5] J. Håstad and K. Risse. On bounded depth proofs for tseitin formulas on the grid; revisited. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1138–1149, 2022. Full version is available at ArXiv:2209.05839 https://arxiv.org/abs/2209.05839.

[6] Jan Krajíček, Pavel Pudlák, and Alan Woods. An exponential lower bound to the size of bounded depth frege proofs of the pigeonhole principle. *Random Structures & Algorithms*, 7(1):15–39, 1995.

[7] Toniann Pitassi, Paul Beame, and Russell Impagliazzo. Exponential lower bounds for the pigeonhole principle. *Computational Complexity*, 3:97–140, 1993. Preliminary version in *STOC '92*.

[8] Toniann Pitassi, Benjamin Rossman, Rocco A. Servedio, and Li-Yang Tan. Poly-logarithmic frege depth lower bounds via an expander switching lemma. In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '16, page 644–657, New York, NY, USA, 2016. Association for Computing Machinery.

[9] G. S. Tseitin. On the complexity of derivation in the propositional calculus. In A. O. Slisenko, editor, *Studies in constructive mathematics and mathematical logic, Part II*, 1968.

Lower Bounds for Regular Resolution Over Parities

Dmitry Itsykson
(joint work with Klim Efremenko, Michal Garlík)

The proof system resolution over parities (Res(⊕)) [7,8] operates with disjunctions of linear equations (linear clauses) over $\mathbb{F}_2$; it extends the resolution proof system by incorporating linear algebra over $\mathbb{F}_2$. Over the years, several exponential lower bounds on the size of tree-like Res(⊕) refutations have been established [1–4, 6, 7, 9–11]. However, proving a superpolynomial lower bound on the size of dag-like Res(⊕) refutations remains a highly challenging open question.

We prove an exponential lower bound for regular Res(⊕). Regular Res(⊕) is a subsystem of dag-like Res(⊕) that naturally extends regular resolution. This is the first known superpolynomial lower bound for a fragment of dag-like Res(⊕) which is exponentially stronger than tree-like Res(⊕). In the regular regime, resolving linear clauses C_1 and C_2 on a linear form f is permitted only if, for both $i \in \{1, 2\}$, the linear form f does not lie within the linear span of all linear forms that were used in resolution rules during the derivation of C_i.

Namely, we show that the size of any regular Res(⊕) refutation of the binary pigeonhole principle BPHP_n^{n+1} is at least $2^{\Omega(\sqrt[3]{n}/\log n)}$. A corollary of our result is an exponential lower bound on the size of a strongly read-once linear branching program solving a search problem. This resolves an open question raised by Gryaznov, Pudlák, and Talebanfard [5].

As a byproduct of our technique, we prove that the size of any tree-like Res(⊕) refutation of the weak binary pigeonhole principle BPHP_n^m is at least $2^{\Omega(n)}$ using Prover-Delayer games. We also give a direct proof of a width lower bound: we show that any dag-like Res(⊕) refutation of BPHP_n^m contains a linear clause C with $\Omega(n)$ linearly independent equations.

References

[1] Paul Beame and Sajin Koroth. On Disperser/Lifting Properties of the Index and Inner-Product Functions. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, volume 251 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 14:1–14:17, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[2] Arkadev Chattopadhyay, Nikhil S. Mande, Swagato Sanyal, and Suhail Sherif. Lifting to parity decision trees via stifling. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge, Massachusetts, USA*, volume 251 of *LIPIcs*, pages 33:1–33:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.

[3] Michal Garlík and Leszek Aleksander Kołodziejczyk. Some subsystems of constant-depth frege with parity. *ACM Trans. Comput. Log.*, 19(4):29:1–29:34, 2018.

[4] Svyatoslav Gryaznov. Notes on resolution over linear equations. In René van Bevern and Gregory Kucherov, editors, *Computer Science - Theory and Applications - 14th International Computer Science Symposium in Russia, CSR 2019, Novosibirsk, Russia, July 1-5, 2019, Proceedings*, volume 11532 of *Lecture Notes in Computer Science*, pages 168–179. Springer, 2019.

[5] Svyatoslav Gryaznov, Pavel Pudlák, and Navid Talebanfard. Linear branching programs and directional affine extractors. In Shachar Lovett, editor, *37th Computational Complexity Conference, CCC 2022, July 20-23, 2022, Philadelphia, PA, USA*, volume 234 of *LIPIcs*, pages 4:1–4:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.
[6] Dmitry Itsykson and Artur Riazanov. Proof complexity of natural formulas via communication arguments. In Valentine Kabanets, editor, *36th Computational Complexity Conference, CCC 2021, July 20-23, 2021, Toronto, Ontario, Canada (Virtual Conference)*, volume 200 of *LIPIcs*, pages 3:1–3:34. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.
[7] Dmitry Itsykson and Dmitry Sokolov. Lower bounds for splittings by linear combinations. In Erzsébet Csuhaj-Varjú, Martin Dietzfelbinger, and Zoltán Ésik, editors, *Mathematical Foundations of Computer Science 2014 - 39th International Symposium, MFCS 2014, Budapest, Hungary, August 25-29, 2014. Proceedings, Part II*, volume 8635 of *Lecture Notes in Computer Science*, pages 372–383. Springer, 2014.
[8] Dmitry Itsykson and Dmitry Sokolov. Resolution over linear equations modulo two. *Ann. Pure Appl. Log.*, 171(1), 2020.
[9] Erfan Khaniki. On proof complexity of resolution over polynomial calculus. *ACM Trans. Comput. Log.*, 23(3):16:1–16:24, 2022.
[10] Jan Krajicek. Randomized feasible interpolation and monotone circuits with a local oracle. *J. Math. Log.*, 18(2):1850012:1–1850012:27, 2018.
[11] Fedor Part and Iddo Tzameret. Resolution with counting: Dag-like lower bounds and different moduli. *Comput. Complex.*, 30(1):2, 2021.

Some Applications of Sunflowers

Dmitry Sokolov

Sunflowers is an extremely powerful object that is widely used in theoretical computer science. The original notion was defined by Erdős, Rado [2].

Definition 1. (k, ℓ)*-sunflower:*

- $S_1, S_2, S_3, \ldots, S_\ell \subseteq \{0,1\}^n$ *of size* k*;*
- $Z \coloneqq \bigcap S_i$*;*
- $\forall i, j \ S_i \cap S_j = Z$*.*

And recently generalization of it was considered by Rossman [5].

Definition 2. (p, ε)*-robust sunflower:*

- $S_1, S_2, S_3, \cdots \subseteq \{0,1\}^n$ *of size* k*;*
- $Z \coloneqq \bigcap S_i$*;*
- $\Pr_{W \sim \mathbf{U}_p}[\exists i, W \subseteq (S_i \setminus Z)] \geq 1 - \varepsilon$*.*

At first viewing, it is not clear why *robust sunflowers* is the more general notion of usual sunflowers. However, through applications of sunflowers, one can note that properties that we typically *want* from sunflowers are exactly what we see in the definition of robust version. In this talk, we will try to show it and discuss the following questions:

- For which problems sunflowers and robust sunflowers are useful?
- What is the *spreadness* of a set? Is it useful to think about spreadness instead of sunflowers?

During this talk, we consider the following applications:

- monotone circuit lower bounds for clique [1, 4];
- lower bounds for Res(k)-proofs of random formulas via sunflowers;
- depth-3 circuit lower bounds via sunflowers (simplification of [3]).

References

[1] Noga Alon and Ravi B. Boppana. The monotone circuit complexity of boolean functions. *Comb.*, 7(1):1–22, 1987.

[2] P. Erdős and R. Rado. Intersection theorems for systems of sets. *Journal of the London Mathematical Society*, 35:85–90, 1960.

[3] Mika Göös, Artur Riazanov, Anastasia Sofronova, and Dmitry Sokolov. Top-down lower bounds for depth-four circuits. In *64th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2023, Santa Cruz, CA, USA, November 6-9, 2023*, pages 1048–1055. IEEE, 2023.

[4] Alexander A. Razborov. Lower bounds on the monotone complexity of some boolean functions. *Dokl. Akad. Nauk SSSR*, 281:798–801, 1985.

[5] Benjamin Rossman. The monotone complexity of k-clique on random graphs. *SIAM J. Comput.*, 43(1):256–279, 2014.

Hardness Condensation by Restriction

Mika Göös

(joint work with Ilan Newman, Artur Riazanov, Dmitry Sokolov)

Hardness condensation is a lower-bound technique in boolean function complexity, where one transforms an n-variate problem f of complexity $k \ll n$ into a related problem f' defined over $\Theta(k)$ variables such that the complexity is preserved at $\Theta(k)$. This approach was first introduced by Buresh-Oppenheim and Santhanam [3] in the context of circuit complexity. Later, it was put to concrete use in the context of proof complexity by Razborov [9] and then further developed in [2, 6, 10, 11]. In these works, the function f' was obtained from f by expander-based function composition.

We study hardness condensation by *restriction*, the simplest operation that reduces the number of variables. Our work focuses on two computational measures: query complexity and communication complexity.

Our first result shows that there exists a function with *query complexity* k such that any its restriction that leaves $O(k)$ variables free has query complexity $O(k^{3/4}\mathrm{poly}(\log k))$. The function that exhibits this is constructed using cheat sheets [1].

Randomized communication complexity is generally non-condensable in a very strong sense: Hambardzumyan, Hatami, and Hatami [7] have shown that there exists a 2^n-by-2^n matrix with communication complexity $\Theta(n^{0.9})$ such that all its $2^{n/2}$-by-$2^{n/2}$ submatrices have constant communication complexity.

Our second result shows that condensation is possible for a very important function: sink-of-xor. This function was used by Chattopadhyay, Mande, and

Sherif [4] to refute log approximate rank conjecture (LARC): they show that $2^{\binom{n}{2}}$-by-$2^{\binom{n}{2}}$ matrix describing sink-of-xor has approximate rank $O(n^4)$ and randomized communication complexity $\Omega(n)$. We show that there exists $2^{O(n)}$-by-$2^{O(n)}$ submatrix of sink-of-xor that retains communication cost $\Omega(n)$. On the other hand, we show that every submatrix of this size has approximate rank at most $O(n^3)$, achieving the stronger negation of LARC conjectured by Chattopadhyay, Garg, and Sherif [5].

The main open question that we leave open is whether the deterministic communication complexity can be condensed by restriction. In a concurrent works Hrubes [8] shows that it can be condensed with a polynomial loss, namely that every 2^n-by-2^n matrix with deterministic communication complexity k has a $2^{O(\sqrt{k})}$-by-$2^{O(\sqrt{k})}$ submatrix of deterministic communication complexity $\Omega(\sqrt{k})$. Can we show that some polynomial loss is necessary?

References

[1] Scott Aaronson, Shalev Ben-David, and Robin Kothari, *Separations in query complexity using cheat sheets*, In Proceedings of the 48th Symposium on Teory of Computing (STOC) (2016), 863–876.

[2] Christoph Berkholz and Jakob Nordström, *Supercritical space-width trade-offs for resolution.*, SIAM Journal on Computing **49(1)** (2020) 98–118.

[3] Joshua Buresh-Oppenheim and Rahul Ranthanam, *Making hard problems harder*, In Proceedings of the 21st Conference on Computational Complexity (CCC) (2006), 73–87.

[4] Arkadev Chattopadhyay, Nikhil Mande, and Suhail Sherif, *The log-approximate-rank conjecture is false*, Journal of the ACM, **67(4)** (2020), 1–28.

[5] Arkadev Chattopadhyay, Ankit Garg, and Suhail Sherif, *Towards stronger counterexamples to the log-approximate-rank conjecture*, In Proceedings of the 41st Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS), **213(13)** (2021), 1–16.

[6] Noah Fleming, Toniann Pitassi, and Robert Robere, *Extremely deep proofs*, In Proceedings of the 13th Innovations in Theoretical Computer Science Conference (ITCS) (2022), 70:1–70:23.

[7] Lianna Hambardzumyan, Hamed Hatami, and Pooya Hatami, *A counter-example to the probabilistic universal graph conjecture via randomized communication complexity*, Discrete Applied Mathematics **322** (2022), 117–122.

[8] Pavel Hrubes, *Hard submatrices for non-negative rank and communication complexity*, Technical report, Electronic Colloquium on Computational Complexity (ECCC), 2024.

[9] Aleksander Razborov, *A new kind of tradeoffs in propositional proof complexity*, Journal of the ACM **63(2)** (2016), 1–14.

[10] Aleksander Razborov, *On space and depth in resolution*, Computational Complexity, **27(3)**, (2017), 511–559.

[11] Aleksander Razborov, *On the width of semialgebraic proofs and algorithms*, Mathematics of Operations Research, **42(4)**, (2017), 1106–1134.

Top-Down Lower Bounds for Depth-Four Circuits

Anastasia Sofronova
(joint work with Mika Göös, Artur Riazanov, Dmitry Sokolov)

We present a top-down lower-bound method for depth-4 boolean circuits. In particular, we give a new proof of the well-known result that the parity function requires depth-4 circuits of size exponential in $n^{1/3}$. Our proof is an application of robust sunflowers and block unpredictability.

The working complexity theorist has three main weapons in their arsenal when proving lower bounds against small-depth boolean circuits (consisting of $\wedge$, $\vee$, $\neg$ gates of unbounded fanin). The most wildly successful ones are the *random restriction* method [1,7] and the *polynomial approximation* method [14,19]. The random restriction method, in particular, is applied ***bottom-up***: it starts by analysing the bottom-most layer of gates next to input variables and finds a way to simplify the circuit so as to reduce its depth by one. The third main weapon, which is the subject of this work, is the ***top-down*** method: starting at the top (output) gate we walk down the circuit in search of a mistake in the computation.

It has been an open problem (posed in [9,12]) to prove exponential lower bounds for depth-4 circuits by a top-down argument. We develop such a lower-bound method in this work and use it to prove a lower bound for the parity function. It has been long known using bottom-up methods that the depth-4 complexity of n-bit parity is $2^{\Theta(n^{1/3})}$ [8, 21]. We recover a slightly weaker bound.

Theorem 1. *Every depth-4 circuit computing the n-bit parity requires $2^{n^{1/3-o(1)}}$ gates.*

Our top-down proof of this theorem is a relatively simple application of two known techniques: robust sunflowers [2, 13, 15] and unpredictability from partial information [12, 18, 20], which we generalise to blocks of coordinates (obtaining essentially best possible parameters).

A major motivation for the further development of top-down methods is that the method is, in a precise sense, *complete* for constant-depth circuits, in that it can be used to prove tight lower bounds (up to polynomial factors) for *any* boolean function. The same is not known to hold for the aforementioned bottom-up techniques. For example, there is currently no known bottom-up proof for the depth-3 circuit lower bound that underlies the oracle separation $\mathsf{AM} \not\subseteq \Sigma_2\mathsf{P}$ [4, 11, 16]. We suspect more generally that top-down methods could prove useful in settings where the bottom-up methods have failed so far, such as proving lower bounds against $\mathsf{AC}^0 \circ \oplus$ circuits computing inner-product [5, 6, 10, 17] or against the polynomial hierarchy in communication complexity [3].

References

[1] Ajtai, M. Σ^1_1-Formulae on finite structures. *Annals Of Pure And Applied Logic.* **24**, 1-48 (1983)

[2] Alweiss, R., Lovett, S., Wu, K. & Zhang, J. Improved bounds for the sunflower lemma. *Annals Of Mathematics.* **194** (2021)

[3] Babai, L., Frankl, P. & Simon, J. Complexity Classes in Communication Complexity Theory. *Proceedings Of The 27th Symposium On Foundations Of Computer Science (FOCS)*. pp. 337-347 (1986)
[4] Böhler, E., Glaßer, C. & Meister, D. Error-Bounded Probabilistic Computations Between MA and AM. *Journal Of Computer And System Sciences*. **72**, 1043-1076 (2006)
[5] Cheraghchi, M., Grigorescu, E., Juba, B., Wimmer, K. & Xie, N. AC0 ∘ MOD2 lower bounds for the Boolean Inner Product. *Journal Of Computer And System Sciences*. **97** pp. 45-59 (2018)
[6] Ezra, M. & Rothblum, R. Small Circuits Imply Efficient Arthur-Merlin Protocols. *Proceedings Of The 13th Innovations In Theoretical Computer Science Conference (ITCS)*. **215** pp. 67:1-67:16 (2022)
[7] Furst, M., Saxe, J. & Sipser, M. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*. **17**, 13-27 (1984)
[8] Håstad, J. Computational Limitations for Small Depth Circuits. (MIT,1987)
[9] Håstad, J., Jukna, S. & Pudlák, P. Top-down lower bounds for depth-three circuits. *Computational Complexity*. **5**, 99-112 (1995)
[10] Huang, X., Ivanov, P. & Viola, E. Affine Extractors and AC0-Parity. *Approximation, Randomization, And Combinatorial Optimization. Algorithms And Techniques, APPROX/RANDOM 2022*. **245** pp. 9:1-9:14 (2022)
[11] Ko, K. Separating and collapsing results on the relativized probabilistic polynomial-time hierarchy. *Journal Of The ACM*. **37**, 415-438 (1990)
[12] Meir, O. & Wigderson, A. Prediction from Partial Information and Hindsight, with Application to Circuit Lower Bounds. *Computational Complexity*. **28**, 145-183 (2019)
[13] Rao, A. Coding for Sunflowers. *Discrete Analysis*. **2020** (2020)
[14] Razborov, A. Lower bounds on the size of bounded depth circuits over a complete basis with logical addition. *Mathematical Notes Of The Academy Of Sciences Of The USSR*. **41**, 333-338 (1987)
[15] Rossman, B. The Monotone Complexity of k-Clique on Random Graphs. *SIAM Journal On Computing*. **43**, 256-279 (2014)
[16] Santha, M. Relativized Arthur–Merlin versus Merlin–Arthur Games. *Information And Computation*. **80**, 44-49 (1989)
[17] Servedio, R. & Viola, E. On a special case of rigidity. *Electron. Colloquium Comput. Complex.*. **TR12-144** (2012), https://eccc.weizmann.ac.il/report/2012/144
[18] Smal, A. & Talebanfard, N. Prediction from partial information and hindsight, an alternative proof. *Inf. Process. Lett.*. **136** pp. 102-104 (2018), https://doi.org/10.1016/j.ipl.2018.04.011
[19] Smolensky, R. Algebraic methods in the theory of lower bounds for Boolean circuit complexity. *Proceedings Of The 19th Symposium On Theory Of Computing (STOC)*. (1987)
[20] Viola, E. AC0 Unpredictability. *ACM Trans. Comput. Theory*. **13**, 5:1-5:8 (2021), https://doi.org/10.1145/3442362
[21] Yao, A. Separating the polynomial-time hierarchy by oracles. *26th Annual Symposium On Foundations Of Computer Science (SFCS)*. (1985)

The Kikuchi Matrix Method

Pravesh Kothari

A 3-SAT formula is a collection of disjunctive 3-clauses (i.e., OR of 3 literals) on a given collection of n truth variables. In the well-known 3-SAT problem, we are given such a 3-SAT formula with m clauses on n variables and our goal is to find an assignment that satisfies all the constraints (if one exists) and if not, find a short (i.e., polynomial size in n) *witness* or *certificate* of unsatisfiability of the formula. 3-SAT is a well-known (and in many a sense, the *first*) NP-complete problem. It is

also a problem that turns out to be hard to approximate. In a more fine-grained picture, denser instances of 3-SAT (i.e., when m grows super-linearly in n) appear intuitively easier (more "easily accessible" information about the satisfying assignment, in the form of additional clauses, if there is one or more "likelihood" of a short contradiction when there are more clauses) but this ease only amounts to an asymptotic gain for formulas with $\omega(n^2)$ constraints. Specifically, we know a $2^{O(n^{1-\delta})}$ time algorithm to find an assignment that gets within $(1-\epsilon)$ factor of the optimal (along with a certificate of approximate optimality) if the formula has at least $\tilde{O}(n^{2+\delta})$ constraints and a polynomial time algorithm if the formula has at least $O(n^3)$ constraints. Back in the late 1980s, in the context of proof complexity, researchers [5] posed the question of whether *random* 3-SAT formulas could be easier than the worst-case. Such formulas are unsatisfiable with high probability if $m \geq O(n)$. Indications of comparative easiness of such formulas finally arrived with the work of Goerdt and Krivilevich [9] and Coja-Oghlan, Goerdt and Lanka [6] in 2004 who proved that random 3-SAT formulas with $\tilde{O}(n^{1.5})$ clauses admit efficient *refutation* algorithms, i.e., polynomial time algorithms that generate a certificate of unsatisfiability of the given formula. And about a decade later, Raghavendra, Rao and Schramm [16], building on the work of Allen, O'Donnell and Witmer [2] proved that there is a $2^{n^{1-\delta}}$ time algorithm to find certificates of unsatisfibility with high probability for formulas with at least $\tilde{O}(n^{1.5-\delta/2})$ clauses. To top this work off, while we lack tools for proving NP-hardness of such *average-case* problems, there are lower bounds in various restricted models [15] (e.g., the sum-of-squares hierarchy of convex relaxations) that show that the running time vs clause density trade-offs achieved in the above works are nearly tight. Finally, all the above story extends naturally to k-SAT (and in fact, all constraint satisfaction problems that generalize k-SAT) for any constant $k \in \mathbb{N}$ with the two relevant threshold values of m being $\tilde{O}(n^{k/2})$ and $\tilde{O}(n^{1+(1-\delta)(k/2-1)})$.

Random k-SAT formulas appear a lot easier than their worst-case counterparts. But could this ease simply be a quirk of the specific random model? Said differently, how "robust" are our conclusions (and our algorithms) with respect to the specific, and rather arbitrary, choice of the random model for the formulas? Such questions [8] were posed in pioneering works of Blum and Spencer and later Feige and Kilian in the 1990s for graph problems. In 2007, Feige [7] proposed a *semi-random* model to formally tackle this question for k-SAT. Feige's goal was to pose a model where an instance is chosen by a combination of random and worst-case choices. The random choices will hopefully steer clear of the worst-case hard formulas while the worst-case component would, in principle, prevent overfitting to specific, brittle properties of a specific random model. Formally, he proposed the *smoothed model* of j-SAT where a formula is chosen by 1) starting with an arbitrary, worst-case k-SAT formula, and, 2) perturbing each literal pattern (i.e., negation pattern on each literal appearing in every clause) independently with some small constant probability, say, 0.1. If the number of clauses $m \geq O(n)$ then such a formula is unsatisfiable with high probability no matter what formula we

begin with. Feige asked the question of whether such smoothed k-SAT formulas admit efficient refutation algorithms and in particular, are they easier than worst-case and in fact, as easy as random k-SAT formulas?

The algorithms that work for random k-SAT formulas strongly exploit the randomness in the variables appearing in the clauses – an aspect completely lost in the smoothed model where the only randomness is the random perturbation of worst-case literal patterns that we begin with. Nevertheless, he managed to find new combinatorial techniques that, when combined with some spectral methods allow *weak*[1] refutation algorithms for such smoothed 3-SAT formulas. These ideas, however, did not yield strong refutation algorithms for 3-SAT and did not generalize to k-SAT for any $k \geq 4$.

In this talk, we presented recent progress and some surprising applications thereof on Feige's smoothed model. In a joint work with Abascal and Guruswami [1], we found *strong* refutation algorithms for smoothed k-SAT formulas with $\tilde{O}(n^{k/2})$ clauses based on new combination of combinatorial and spectral methods. These results were then generalized to obtain the same running time vs clause density trade-off (i.e., $2^{n^{1-\delta}}$ time for formulas with $\tilde{O}(n^{1+(1-\delta)(k/2-1)})$ clauses) in a later joint work with Manohar and Guruswami [10] based on a new tool called *Kikuchi matrices* combined with a new *regularity decomposition* for hypergraphs. Simpler proof was later found in a joint work with Hsieh and Mohanty [11] and with Munha-Correia and Sudakov [12].

Somewhat surprisingly, these new algorithms have applications to problems in combinatorics and coding theory that we also discussed in the talk. The principle behind these applications is simple if somewhat strange. In principle, the truth of any mathematical statement can be efficiently encoded into a satisfiability of a 3-SAT formula thus reducing a mathematical problem to understanding whether the formula produced by the reduction is satisfiable. This abstract idea, however, is too general to be useful as a tool for actually establishing mathematical results. In our applications, however, we'd be able to encode the truth of certain kinds of combinatorial statements as the satisfiability of a *family of* SAT formulas and thus, to disprove the truth of such a statement, it is enough to prove that one of these formulas, say a *randomly* chosen member, is unsatisfiable. While this may appear to get us closer to random SAT formulas, the resulting formulas are far from random. In fact, in a precise sense, they can be described by a number of random bits that is significantly smaller (in applications n^{ϵ} for $\epsilon \ll 1$ or even $\mathsf{poly}\log n$) than the number of variables that disallows straightforward probabilistic analyses. Nevertheless, it turns out that the *analysis of the refutation algorithms for smoothed formulas* above can be adapted with some work to apply to even such randomness-starved formulas. Notice that we do not need any efficient algorithm

[1] A weak refutation algorithm certifies unsatisfiability of a 3-SAT formula, as opposed to a strong refutation algorithm that certifies that the every assignment must violate a constant fraction of the clauses in the input formula. The results discussed for random 3-SAT above all yield strong refutation algorithms.

for proving unsatisfiability of the SAT formula in such an application. The algorithm arises purely as a tool for arguing the unsatisfiability (indeed, we know of no other proofs, in general, for establishing such a result).

The applications of this technique so far include a new cubic (improving on the quadratic) lower bounds on the blocklength of a 3-query locally decodable codes [3], exponential (improving on cubic) lower bounds [13] on the block length of 3-query, linear, locally correctable codes (with applications to almost resolving the Hamada conjecture from the theory of algebraic designs for 4-designs), a super-polynomial lower bound [14] for non-linear 3-query locally correctable codes, the resolution of Feige's conjecture [10] on the hypergraph Moore bound, and improved bounds on three-term arithmetic progressions with random common differences [3, 4]. In the talk, we focused largely on the lower bounds on the local codes.

References

[1] J. Abascal, V. Guruswami, and P. K. Kothari. Strongly refuting all semi-random boolean csps. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10 - 13, 2021*, pages 454–472. SIAM, 2021.

[2] S. R. Allen, R. O'Donnell, and D. Witmer. How to refute a random CSP. *CoRR*, abs/1505.04383, 2015.

[3] O. Alrabiah, V. Guruswami, P. K. Kothari, and P. Manohar. A near-cubic lower bound for 3-query locally decodable codes from semirandom CSP refutation. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC 2023, Orlando, FL, USA, June 20-23, 2023*, pages 1438–1448. ACM, 2023.

[4] J. Briët and D. Castro-Silva. On the threshold for szemerédi's theorem with random differences, 2023.

[5] V. Chvátal and E. Szemerédi. Many hard examples for resolution. *J. ACM*, 35(4):759–768, 1988.

[6] A. Coja-Oghlan, A. Goerdt, and A. Lanka. Strong refutation heuristics for random k-sat. In *8th International Workshop on Randomization and Computation, RANDOM 2004, Cambridge, MA, USA, August 22-24, 2004, Proceedings*, volume 3122 of *Lecture Notes in Computer Science*, pages 310–321. Springer, 2004.

[7] U. Feige. Refuting smoothed 3cnf formulas. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007), October 20-23, 2007, Providence, RI, USA, Proceedings*, pages 407–417. IEEE Computer Society, 2007.

[8] U. Feige. Introduction to semirandom models. In *Beyond the Worst-Case Analysis of Algorithms*, pages 189–211. Cambridge University Press, 2020.

[9] A. Goerdt and M. Krivelevich. Efficient recognition of random unsatisfiable k-sat instances by spectral methods. In *STACS 2001, 18th Annual Symposium on Theoretical Aspects of Computer Science, Dresden, Germany, February 15-17, 2001, Proceedings*, volume 2010 of *Lecture Notes in Computer Science*, pages 294–304. Springer, 2001.

[10] V. Guruswami, P. K. Kothari, and P. Manohar. Algorithms and certificates for boolean CSP refutation: smoothed is no harder than random. In *STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 - 24, 2022*, pages 678–689. ACM, 2022.

[11] J. Hsieh, P. K. Kothari, and S. Mohanty. A simple and sharper proof of the hypergraph moore bound. In *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, pages 2324–2344. SIAM, 2023.

[12] Jun-Ting Hsieh, Pravesh K. Kothari, Sidhanth Mohanty, David Munhá Correia, and Benny Sudakov. Small even covers, locally decodable codes and restricted subgraphs of edge-colored Kikuchi graphs, 2024.

[13] P. K. Kothari and P. Manohar. An exponential lower bound for linear 3-query locally correctable codes. *CoRR*, abs/2311.00558, 2023.
[14] P. K. Kothari and P. Manohar. Superpolynomial lower bounds for smooth 3-lccs and sharp bounds for designs. *FOCS*, 2024.
[15] P. K. Kothari, R. Mori, R. O'Donnell, and D. Witmer. Sum of squares lower bounds for refuting any CSP. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 132–145. ACM, 2017.
[16] P. Raghavendra, S. Rao, and T. Schramm. Strongly refuting random csps below the spectral threshold. *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 121–131, 2017.

Decoding Codes via Proofs

Madhur Tulsiani

The problem of finding the nearest codeword to a possibly corrupted received word, can naturally be viewed as an optimization problem. Over the past few years, approaching this optimization problem via continuous relaxations have led to new unique decoding and list decoding algorithms for several code families.

This talk discussed a general framework for obtaining such algorithms using relaxations based on the Sum-of-Squares (SoS) hierarchy of semidefinite programs. In particular, this framework is an adaptation of the well-known "proofs to algorithms" paradigm to the setting of codes [1]. If the proof of the fact that all pairs of codewords have large distance can be expressed in a (positivestellensatz) proof system corresponding to the SoS hierarchy, then one can use it to obtain a list decoding algorithm for the corresponding code. This can easily be seen to be the case for several code families based on expander graphs, where the proofs of distance rely on spectral properties of these graphs [2, 3]. Using the fact that spectral bounds can be phrased as SoS inequalities, this yields the first efficient list-decoding algorithms for several such families of codes.

References

[1] Noah Fleming, Pravesh Kothari, and Toniann Pitassi. Semialgebraic proofs and efficient algorithm design. *Foundations and Trends® in Theoretical Computer Science*, 14(1-2):1–221, 2019.
[2] M. Sipser and D. Spielman. Expander codes. *IEEE Transactions on Information Theory*, 42(6):1710–1722, 1996. Preliminary version in *Proc. of FOCS'94*.
[3] Amnon Ta-Shma. Explicit, almost optimal, epsilon-balanced codes. In *Proceedings of the 49th ACM Symposium on Theory of Computing*, STOC 2017, pages 238–251, New York, NY, USA, 2017. ACM.

Graph Homomorphisms and Polynomials

Grigoriy Blekherman

If we are given a polynomial expression in traces of powers of real symmetric matrices, such as

$$3[\mathrm{tr}(A^2)]^2\,\mathrm{tr}(B^4) - 4\,\mathrm{tr}(A^4)[\mathrm{tr}(B^2)]^2,$$

is there an algorithm to decide whether this expression is nonnegative for all real symmetric matrices A, B *of all sizes*? What happens if we replace trace by normalized trace $\widetilde{\mathrm{tr}}(A) = \frac{\mathrm{tr}(A)}{n}$, where n is the size of the matrix?

The first (unnormalized) problem is *undecidable*, while the second one is *decidable*. The key to the hardness of the unnormalized problem is the beautiful geometry of *the image of the probability simplex under the Vandermonde map*.

For any $n \times n$ matrix A recall that $\mathrm{tr}(A^d) = \lambda_1^d + \cdots + \lambda_n^d$, where λ_i are the eigenvalues of A. Let p_d to denote the d-th power sum polynomial: $p_d(x) = x_1^d + \cdots + x_n^d$. Testing whether $3[\mathrm{tr}(A^2)]^2\,\mathrm{tr}(B^4) - 4\,\mathrm{tr}(A^4)[\mathrm{tr}(B^2)]^2$ is nonnegative on all real symmetric matrices of all sizes is equivalent to understanding whether $3p_2^2(x)p_4(y) - 4p_4(x)p_2^2(y)$ is nonnegative on all real vectors x and y of any dimension. Define the *d-th Vandermonde map* $\nu_{n,d}$ by sending a point in $\mathbb{R}^n$ to its image under the first d power sums:

$$\nu_{n,d}(x) = (p_1(x), \ldots, p_d(x)).$$

Let Δ_{n-1} be the probability simplex in $\mathbb{R}^n$: Δ_{n-1} consists of all vectors with nonnegative coordinates with the sum of coordinates equal to 1. The image $\nu_{n,d}(\Delta_{n-1})$ is called the *the (n,d)-Vandermonde cell* and is denoted by $\Pi_{n,d}$. Observe that the first coordinate of $\Pi_{n,d}$ is identically 1, and so we may project it out, and see $\Pi_{n,d}$ as the subset of $\mathbb{R}^{d-1}$, which is the image of Δ_{n-1} under $(p_2, \ldots, p_d)$.

Since all of our exponents are even, and the polynomial $3p_2^2(x)p_4(y) - 4p_4(x)p_2^2(y)$ is bi-homogeneous in x and y, it follows that deciding nonnegativity of $3p_2^2(x)p_4(y) - 4p_4(x)p_2^2(y)$ for all $x, y \in \mathbb{R}^n$ is equivalent to deciding nonnegativity $3b - 4a$ on the product $\Pi_{n,2} \times \Pi_{n,2}$ via the substituion $a = \frac{p_4(x)}{p_2^2(x)}, b = \frac{p_4(y)}{p_2^2(y)}$.

By considering only even exponents, and only multihomogeneous polynomials, we can by taking ratios of the form $\frac{p_{2k}(x)}{p_2^k(x)}$ reduce the problem to understanding the geometry of the Vandermonde cell $\Pi_{n,d}$. The image of $\Pi_{n,3}$ looks as follows:

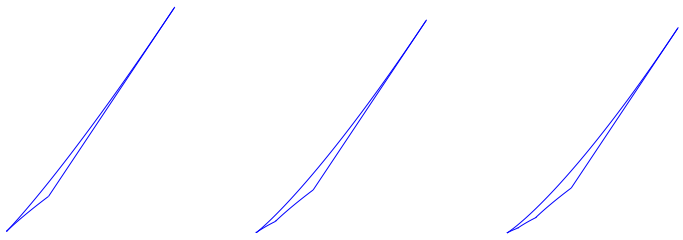

Figure 1. The sets $\Pi_{n,3}$ for $n = 3, 4, 5$

The Vandermonde cell $\Pi_{n,3}$ has n special points of the form $(1/k, 1/k^2)$ for $k = 1, \ldots, n$. The lower boundary of the image consists of concave curves joining the special points $(1/k, 1/k^2)$. As n goes to infinity we get infinitely many isolated points on the curve $y = x^2$. The upper boundary of the limit cell $\Pi_{\infty,3}$ is given by the curve $y^2 = x^3$ for $0 \leq x \leq 1$.

Once we have several matrix variables, we will get one copy of $\Pi_{n,3}$ for every matrix variable, and so we will want to test nonnegativity of polynomials on products $\Pi_{n,3} \times \Pi_{n,3} \times \cdots \times \Pi_{n,3}$. This allows us to prove undecidability using the existing reduction of Hatami and Norin [4], who applied it in the context of proving undecidability of inequalities in graph homomorphism densities. The details are provided in [1], and we describe a connection between geometry of the Vandermonde map and graph homomorphism density inequalities.

Given two simple graphs G, H a homomorphism $\varphi : G \to H$ is a map from the vertex set $V(G)$ of G to the vertex set $V(H)$ of H such that φ preserves adjacency of vertices. The *homomorphism density* $t(H, G)$ is defined by

$$t(H, G) = \frac{\#\text{homomorphisms}\, G \to H}{\text{total } \#\text{ of maps}\, V(G) \to V(H)}.$$

One can ask for complexity of deciding whether a polynomial expression in $t(G_i, H)$ is nonnegative for all target graphs H, and it was shown by Hatami and Norin that this problem is undecidable [4]. We now describe an alternative proof of this result using the geometry of the Vandermonde map.

For a graph H let A_H denote the adjacency matrix of H. It is well known that for an even cycle C_{2k} we have

$$t(C_{2k}, H) = \frac{\operatorname{tr} A_H^{2k}}{|V(H)|^{2k}}.$$

Therefore we have

$$\frac{t(C^6, H)}{t(C_2, H)^3} = \frac{\operatorname{tr} A_H^6}{(\operatorname{tr} A_H^2)^3}, \text{ and } \frac{t(C^4, H)}{t(C_2, H)^2} = \frac{\operatorname{tr} A_H^4}{(\operatorname{tr} A_H^2)^2}.$$

If we do not restrict to adjacency matrices of graphs, then we are simply looking at the Vandermonde cell $\Pi_{n,3}$. However, one can show that as we go over all graphs H with any number of vertices, we actually get the full Vadermonde cell $\Pi_{n,3}$ [2].

However, for undecidability we need a product of independent copies of $\Pi_{n,3}$. We can obtain these independent copies by considering *necklace graphs*. The even cycles are $2k$ copies of complete graph K_2 glued together in a circular fashion. One can similarly take $2k$ copies of the triangle K_3 and glue them together in circular fashion, producing a necklace graph on $4k$ vertices. One can show that using necklace graphs for larger complete graphs K_m we can obtain independent copies of $\Pi_{n,3}$ and use the same undecidability reduction [3] [2].

References

[1] Jose Acevedo, Grigoriy Blekherman, Sebastian Debus, and Cordian Riener. The wonderful geometry of the Vandermonde map. *arXiv preprint arXiv:2303.09512*, 2023.

[2] Grigoriy Blekherman and Annie Raymond. Ubiquity of power sums in graph profiles. *arXiv preprint arXiv:2308.07422*, 2023.

[3] Grigoriy Blekherman, Annie Raymond, and Fan Wei. Undecidability of polynomial inequalities in weighted graph homomorphism densities. *Forum of Mathematics, Sigma*, 12:e40, 2024.

[4] Hamed Hatami and Serguei Norine. Undecidability of linear inequalities in graph homomorphism densities. *Journal of the American Mathematical Society*, 24(2):547–565, 2011.

On the Algebraic Proof Complexity of Tensor Isomorphism

Nicola Galesi

(joint work with Joshua A. Grochow, Toniann Pitassi, Adrian She)

The Tensor Isomorphism problem (TI) has recently emerged as having connections to multiple areas of research within complexity and beyond, but the current best upper bound is essentially the brute force algorithm. Being an algebraic problem, TI (or rather, proving that two tensors are *non*-isomorphic) lends itself very naturally to algebraic and semi-algebraic proof systems, such as the Polynomial Calculus (PC) and Sum of Squares (SoS). For its combinatorial cousin Graph Isomorphism, essentially optimal lower bounds are known for approaches based on PC and SoS [1]. Our main results are an $\Omega(n)$ lower bound on PC degree or SoS degree for Tensor Isomorphism, and a nontrivial upper bound for testing isomorphism of tensors of bounded rank.

We also show that PC cannot perform basic linear algebra in sub-linear degree, such as comparing the rank of two matrices (which is essentially the same as 2-TI), or deriving $BA = I$ from $AB = I$. As linear algebra is a key tool for understanding tensors, we introduce a strictly stronger proof system, PC+Inv, which allows as derivation rules all substitution instances of the implication $AB = I \to BA = I$. We conjecture that even PC+Inv cannot solve TI in polynomial time either, but leave open getting lower bounds on PC+Inv for any system of equations, let alone those for TI. We also highlight many other open questions about proof complexity approaches to TI.

References

[1] Christoph Berkholz and Martin Grohe. Linear Diophantine equations, group CSPs, and graph isomorphism. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 327–339. SIAM, Philadelphia, PA, 2017.

Announcing Tropical Proof Systems

Edward A. Hirsch
(joint work with Yaroslav Alekseev, Dima Grigoriev)

Tropical (min-plus) arithmetic has many applications in various areas of mathematics. The operations are the real addition (as the tropical multiplication) and the minimum (as the tropical addition). Recently, [1, 4, 6] demonstrated a version of Nullstellensatz in the tropical setting.

In this talk we introduce "tropical proof systems": (semi)algebraic proof systems that use min-plus arithmetic. This allows us to view some known proof systems from a different angle. In particular, we provide a static Nullstellensatz-based tropical proof system `MP-NS` that (equipped with dual Boolean variables) polynomially simulates daglike resolution and also has short proofs for the propositional pigeon-hole principle. Its dynamic version strengthened by an additional derivation rule (a tropical analogue of resolution by linear inequality) is equivalent to the system `Res(LP)` [5], which derives nonnegative linear combinations of linear inequalities; this latter system is known to polynomially simulate Krajíček's`Res(CP)` [7] with unary coefficients. No exponential lower bounds are known for this system; there are recent results [2, 3] for a treelike version only. For the truth values encoded by $\{0, \infty\}$, dynamic tropical proofs are equivalent to `Res(`∞`)`, which is a small-depth Frege system called also DNF resolution.

Therefore, tropical proof systems give a finer hierarchy of proof systems below `Res(LP)` for which we still do not have exponential lower bounds. For the weakest of them, `MP-NS` mentioned above, we can prove an exponential lower bound for a non-CNF (and very simple) system of inequalities (it expresses that a large tropical power of a Boolean variable equals a non-Boolean constant). The method of proving the bound is also simple enough: we construct a directed graph on monomials occurring in a tropical algebraic combination and analyze the coefficients of the algebraic combination this way. Therefore, we hope for new superpolynomial lower bounds for tropical proof systems of intermediate power.

The new notion of a tropical proof system leaves multiple open questions and directions for further research.

References

[1] Aaron Bertram and Robert Easton. The tropical Nullstellensatz for congruences. *Adv. Math*, 308:36–82, 2017.

[2] Noah Fleming, Mika Göös, Russell Impagliazzo, Toniann Pitassi, Robert Robere, Li-Yang Tan, and Avi Wigderson. On the power and limitations of branch and cut. In Valentine Kabanets, editor, *36th Computational Complexity Conference, CCC 2021, July 20-23, 2021, Toronto, Ontario, Canada (Virtual Conference)*, volume 200 of *LIPIcs*, pages 6:1–6:30. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[3] Max Gläser and Marc E. Pfetsch. Sub-exponential lower bounds for branch-and-bound with general disjunctions via interpolation. Technical Report 2308.04320, arXiv, 2023.

[4] Dima Grigoriev and Vladimir V. Podolskii. Tropical effective primary and dual Nullstellensätze. *Discrete and Computational Geometry*, 59(3):507–552, 2018.

[5] Edward A. Hirsch and Arist Kojevnikov. Several notes on the power of Gomory-Chvátal cuts. *Ann. Pure Appl. Log.*, 141(3):429–436, 2006.
[6] Daniel Joo and Kalina Mincheva. Prime congruences of additively idempotent semirings and a Nullstellensatz for tropical polynomials. *Selecta Math.*, 24:2207–2233, 2018.
[7] Jan Krajícek. Discretely ordered modules as a first-order extension of the cutting planes proof system. *J. Symb. Log.*, 63(4):1582–1596, 1998.

Meta-Mathematics of Complexity Theory

Igor C. Oliveira

Despite significant efforts from computer scientists and mathematicians, the P vs. NP problem and other fundamental questions about the complexity of computations seem to remain out of reach for existing techniques. The difficulty of making progress on such problems has motivated a number of researchers to investigate the logical foundations of computational complexity. Over the last few decades, several works at the intersection of logic and complexity theory showed that certain fragments of Peano Arithmetic collectively known as Bounded Arithmetic (see, e.g., [6, 7]) can formalize a large fraction of results from algorithms and complexity (e.g., the PCP Theorem [11] and complexity lower bounds against restricted classes of Boolean circuits [10]). It is natural to consider if the same theories can settle longstanding problems about the inherent difficulty of computations.

In the first part of this talk, we survey a few recent results [1–4, 8, 9, 12] on the unprovability of statements of interest to complexity theory in theories of Bounded Arithmetic and highlight some open problems. In the second part of the talk, we will cover new results on the reverse mathematics of complexity lower bounds [5], a research direction which seeks to determine which axioms are necessary to prove certain results. We explore reversals in the setting of bounded arithmetic, with Cook's theory PV_1 as the base theory, and show that several natural lower bound statements about communication complexity, error correcting codes, and Turing machines are equivalent to widely investigated combinatorial principles such as the weak pigeonhole principle for polynomial-time functions and its variants. As a consequence, complexity lower bounds can be formally seen as fundamental mathematical axioms with far-reaching implications. Time permitting, we will also present several implications of these results:

- Under a plausible cryptographic assumption, the classical single-tape Turing machine $\Omega(n^2)$-time lower bound for Palindrome is unprovable in Jerábek's theory APC_1.
- While APC_1 proves one-way communication lower bounds for Set Disjointness, it does not prove one-way communication lower bounds for Equality, under a plausible cryptographic assumption.
- An amplification phenomenon connected to the (un)provability of some lower bounds, under which a quantitatively weak $n^{1+\varepsilon}$ lower bound is provable if and only if a stronger (and often tight) n^c lower bound is provable.

- Feasibly definable randomized algorithms can be feasibly defined deterministically (APC_1 is $\forall\Sigma^b_1$-conservative over PV_1) if and only if one-way communication complexity lower bound for Set Disjointness are provable in PV_1.

References

[1] Albert Atserias, Sam Buss, and Moritz Müller. On the consistency of circuit lower bounds for non-deterministic time. In *Symposium on Theory of Computing* (STOC), pages 1257–1270, 2023.

[2] Jan Bydzovsky, Jan Krajíček, and Igor C. Oliveira. Consistency of circuit lower bounds with bounded theories. *Logical Methods in Computer Science*, 16(2), 2020.

[3] Jan Bydzovsky and Moritz Müller. Polynomial time ultrapowers and the consistency of circuit lower bounds. *Arch. Math. Log.*, 59(1-2):127–147, 2020.

[4] Marco Carmosino, Valentine Kabanets, Antonina Kolokolova, and Igor C. Oliveira. Learn-uniform circuit lower bounds and provability in bounded arithmetic. In *Symposium on Foundations of Computer Science* (FOCS), 2021.

[5] Lijie Chen, Jiatu Li, and Igor C. Oliveira. Reverse mathematics of complexity lower bounds. *Electronic Colloquium on Computational Complexity* (ECCC), TR:24:060, 2024.

[6] Stephen A. Cook and Phuong Nguyen. *Logical Foundations of Proof Complexity*. Cambridge University Press, 2010.

[7] Jan Krajíček. *Bounded Arithmetic, Propositional Logic, and Complexity Theory*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1995.

[8] Jan Krajíček and Igor C. Oliveira. Unprovability of circuit upper bounds in Cook's theory PV. *Logical Methods in Computer Science*, 13(1), 2017.

[9] Jiatu Li and Igor C. Oliveira. Unprovability of strong complexity lower bounds in bounded arithmetic. In *Symposium on Theory of Computing* (STOC), 2023.

[10] Moritz Müller and Ján Pich. Feasibly constructive proofs of succinct weak circuit lower bounds. *Annals of Pure and Applied Logic*, 171(2), 2020.

[11] Ján Pich. Logical strength of complexity theory and a formalization of the PCP theorem in bounded arithmetic. *Logical Methods in Computer Science*, 11(2), 2015.

[12] Ján Pich and Rahul Santhanam. Strong co-nondeterministic lower bounds for NP cannot be proved feasibly. In *Symposium on Theory of Computing* (STOC), pages 223–233, 2021.

On the Theory of Exponential Integer Parts

Emil Jeřábek

An *integer part (IP)* of an ordered ring R is a discretely ordered subring $I \subseteq R$ such that every $x \in R$ is within distance 1 from I. (By abuse of language, we will conflate a discretely ordered ring I with the ordered semiring $I_{\geq 0}$.) A classical result of Shepherdson [6] characterizes models of IOpen (= Robinson's arithmetic + induction for open formulas in the language $\mathcal{L}_{\mathrm{OR}} = \langle 0, 1, +, \cdot, < \rangle$):

Theorem 1. *Integer parts of real-closed fields are exactly the models of* IOpen.

Let an *exponential field* be an ordered field R endowed with an isomorphism $\exp\colon \langle R, 0, 1, +, < \rangle \to \langle R_{>0}, 1, 2, \cdot, < \rangle$, optionally satisfying the *growth axiom (GA)* $\exp(x) > x$. Introduced by Ressayre [5], an *exponential integer part (EIP)* of an exponential ordered field $\langle R, \exp \rangle$ is an IP $I \subseteq R$ such that $I_{\geq 0}$ is closed under exp. We are interested in the question of characterizing (non-negative parts of) ordered

rings that are EIP of real-closed exponential fields (RCEF), and in particular, what is the first-order theory of such rings. This problem (and in particular, the question whether this theory properly extends IOpen) was raised by Jeřábek [2], who provided an upper bound: all countable models of the bounded arithmetical theory VTC^0 in $\mathcal{L}_{\mathrm{OR}}$ are EIP of RCEF.

Extensions of Theorem 1 to exponential ordered fields were previously studied by Boughattas and Ressayre [1] and Kovalyov [4], but they focused on generalizing the other direction of the theorem (e.g., what additional properties of RCEF ensure that their EIP are models of open induction in a language with exponentiation?). Moreover, they were mostly concerned with EIP in a language with the binary powering operation $x^y = \exp(y \log x)$. Since $\langle I, +, \cdot, <, x^y\rangle$ can define approximations of exp on its fraction field F, we can canonically extend exp to the completion of F; but no such direct construction seems possible for EIP in $\mathcal{L}_{\mathrm{OR}}$ or $\mathcal{L}_{\mathrm{OR}} \cup \{2^x\}$, hence our arguments will be of different nature.

The main goal of this talk is to present complete axiomatizations of the first-order theories of EIP of RCEF in $\mathcal{L}_{\mathrm{OR}} \cup \{2^x\}$, $\mathcal{L}_{\mathrm{OR}} \cup \{P_2\}$ (where P_2 is a predicate for the image of 2^x), and $\mathcal{L}_{\mathrm{OR}}$, denoted TEIP_{2^x}, TEIP_{P_2}, and TEIP, and to determine some basic properties of these theories.

The theories TEIP_{2^x} and TEIP_{P_2} are axiomatized over IOpen by a finite list of a few obvious axioms. The theory TEIP is more involved: it has an infinite schema of axioms PWin^0_n expressing, for each $n \in \mathbb{N}$, that the second player has a winning strategy in the *power-of-two game* PowG_n. This game is played between two players, Challenger (C) and Powerator (P), in n rounds: in round $0 \le i < n$, C picks $x_i > 0$, and P responds with $u_i > 0$ such that $u_i \le x_i < 2u_i$. C wins if $u_i u_j < u_h < 2u_i u_j$ for some $h, i, j < n$, otherwise P wins. The motivation for the game is that if $\langle \mathfrak{M}, P_2\rangle \vDash \mathsf{TEIP}_{P_2}$, then "play $u_i \in P_2$" is a winning strategy for P.

Using the existence of a nonstandard model of IOpen that is a UFD (Smith [7]), we can show that TEIP properly extends IOpen.

The main problem about basic properties of TEIP is whether it is finitely axiomatizable over IOpen. As a partial progress, we show that the formulas $\mathrm{PWin}^1_n(u)$, $n \in \mathbb{N}$, form a strictly increasing hierarchy over $\mathrm{Th}(\mathbb{N})$, where $\mathrm{PWin}^1_n(u)$ expresses that P wins the game $\mathrm{PowG}^1_n(u)$ defined like PowG_n, but with the first move of P fixed as u. To this end, we prove reasonably tight upper and lower bounds on the complexity measure $c(u) = \min\{n : \text{C wins } \mathrm{PowG}^1_{n+1}(u)\}$, showing that $\{c(u) : u \text{ not a power of } 2\} = \mathbb{N}_{>0}$. For example, $k + 1 \le c\big(6^{2^{2^k}!}\big) \le k + 4$.

These bounds also imply that there are models $\langle \mathfrak{M}, P_2\rangle \vDash \mathrm{Th}(\mathbb{N}) + \mathsf{TEIP}_{P_2}$ such that P_2 is incomparable with the set of "oddless numbers" (i.e., whose all nontrivial divisors are even); indeed, $u \in P_2$ may even be divisible by 3.

This talk is based on [3]. The work was supported by the Czech Academy of Sciences (RVO 67985840) and GA ČR project 23-04825S.

References

[1] S. Boughattas and J.-P. Ressayre, *Arithmetization of the field of reals with exponentiation extended abstract*, RAIRO – Theoretical Informatics and Applications **42** (2008), 105–119.

[2] E. Jeřábek, *Models of* VTC^0 *as exponential integer parts*, Mathematical Logic Quarterly **69** (2023), 244–260.

[3] ———, *On the theory of exponential integer parts*, arXiv:2404.06888 [math.LO], 2024, `https://arxiv.org/abs/2404.06888`.

[4] K. Kovalyov, *Analogues of Shepherdson's Theorem for a language with exponentiation*, arXiv:2306.02012 [math.LO], 2023, `https://arxiv.org/abs/2306.02012`.

[5] J.-P. Ressayre, *Integer parts of real closed exponential fields*, in: Arithmetic, proof theory, and computational complexity (P. Clote and J. Krajíček, eds.), Oxford Logic Guides vol. 23, Oxford University Press, 1993, 278–288.

[6] J. C. Shepherdson, *A nonstandard model for a free variable fragment of number theory*, Bulletin de l'Académie Polonaise des Sciences, Série des Sciences Mathématiques, Astronomiques et Physiques **12** (1964), 79–86.

[7] S. T. Smith, *Building discretely ordered Bezout domains and GCD domains*, Journal of Algebra **159** (1993), 191–239.

Quantified Propositional Calculi and Narrow Implicit Proofs

Pavel Pudlák

Quantified Propositional Calculus G**.** This is the sequent calculus that uses quantified propositions and all rules, including quantifier rules [2]. We have to specify how the quantifier rules are used because in propositional calculus there are no terms. In this paper we will use quantifier-free formulas in place of terms in LK in the quantifier rules.

Fragments G_i **of** G**.** The classes Σ^q_i, Π^q_i of quantified propositional formulas are defined in the usual way. For $i \geq 1$, G_i denotes the Σ^q_i fragment of G, which is G restricted to formulas of Σ^q_i.

Implicit proofs and narrow implicit proofs. Jan Krajíček [1] defined an operation that from two proof systems P and Q produces another proof system, which he denoted by $[P, Q]$. The proof system $[P, Q]$ can be roughly described as follows. A proof of ϕ in $[P, Q]$ is a pair (Π, C), where C is a Boolean circuit that succinctly defines a possibly exponential size Q-proof of ϕ, and Π is a P-proof of a formula that expresses this fact. In the special case when $P = Q$, Krajíček calls such a proof system *implicit* P and denotes it by iP. For natural proof systems P, this operation seems to produce from P an essentially stronger proof system iP.

Krajíček also defined a restricted version of $[P, Q]$ and denoted it by $[P, Q]^m$; we will call it *the narrow implicit proof system defined by* P, Q. Unlike the general concept of implicit proofs, narrow implicit proof systems are only defined when the proof system Q is based on formulas.

Definition 1. *Let* P *be an arbitrary proof system and* Q *a formula based proof system with a class of formulas* $\mathcal{F}$ *and a set of deduction rules* $R_1, \ldots, R_l$*. A proof of a formula* ϕ *in* $[P, Q]^m$ *(the* narrow implicit proof system *defined by* P, Q*) is a pair* (Π, C) *where* C *is a circuit computing a Boolean function* $f_C : \{0, 1\}^n \to$

$\{0,1\}^m$ and Π is a P-proof of the formula $\gamma_{C,\phi}$ which says that f_C correctly encodes a Q-proof of ϕ. In more detail, $\gamma_{C,\phi}$ should express the following condition:

- *For an $i \in \{0,1\}^n$, $C(i)$, the bit-string that C outputs, encodes a string consisting of a formula and $k+1$ numbers $(\phi_i; i_1, \ldots, i_k; j)$ such that $i_1, \ldots, i_k < i$ and ϕ_i follows from $\phi_{i_1}, \ldots, \phi_{i_k}$ using rule R_j;*

where we view strings in $\{0,1\}^n$ as numbers in the interval $[0, 2^n - 1]$; for formalizing C in the propositional calculus, we use variables for every vertex of C and clauses expressing that the values correspond to the gates at the vertices.

We denote by *Res* the resolution proof system.

Theorem 1. *$[Res, G_i]^m$ is polynomially equivalent to G_{i+1} for $i \geq 1$, i.e., the two proof systems polynomially simulate each other.*

The existence of the simulation of $[Res, G_i]^m$ by G_{i+1} is proved by proving soundness of $[Res, G_i]^m$ in T_2^i. It is well-known that provability of the soundness of a proof system P in T_2^i implies the existence of a polynomial simulation of P by G_i, see [2].

The opposite simulation is based on cut-elimination. Given a proof Π in G_{i+1}, we eliminate all cuts with Σ^q_{i+1} formulas. Thus the resulting proof Π' is a proof in G_i. The proof has exponential size, if one uses substitutions instead of repeating parts of Π, and can be succinctly defined by a polynomial size circuit. To make an implicit proof from Π', one has to solve a number of technical problems.

References

[1] J. Krajíček: Implicit proofs, J. of Symbolic Logic, 69(2), (2004), pp.387-397.

[2] J. Krajíček and P. Pudlák: "Quantified Propositional Calculi and Fragments of Bounded Arithmetic", Zeitschr. f. Mathematikal Logik u. Grundlagen d. Mathematik, Bd. 36(1), (1990), pp. 29-46.

Compressing CFI Graphs and Lower Bounds for the Weisfeiler–Leman Refinements

Martin Grohe

(joint work with Moritz Lichter, Daniel Neuen, and Pascal Schweitzer)

The k-dimensional Weisfeiler-Leman (k-WL) algorithm is a simple combinatorial algorithm that was originally designed as a graph isomorphism heuristic. It naturally finds applications in Babai's quasipolynomial-time isomorphism algorithm, practical isomorphism solvers, and algebraic graph theory. However, it also has surprising connections to other areas such as logic, combinatorial optimization, machine learning, and proof complexity.

The algorithm iteratively computes a coloring of the k-tuples of vertices of a graph. Since Fürer's linear lower bound [1], it has been an open question whether there is a super-linear lower bound for the iteration number for k-WL on graphs.

We answer this question affirmatively, establishing an $\Omega(n^{k/2})$-lower bound for all k.

References

[1] Martin Fürer. Weisfeiler-Lehman refinement requires at least a linear number of iterations. In *Automata, languages and programming*, volume 2076 of *Lecture Notes in Comput. Sci.*, pages 322–333. Springer, Berlin, 2001.

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

Shuo Pang

(joint work with Duri Janett, Jakob Nordström)

We study trade-offs in proof complexity and Weisfeiler–Leman algorithms. In a trade-off between a pair of complexity measures, if the first measure is constrained to be small, then usually, the lower bound on the second measure stays below the trivial worst-case upper bound. By contrast, in a so-called supercritical trade-off, the lower bound on the second measure is larger than its worst-case upper bound (see e.g. [1, 3–6, 8]). We present the first resolution width-depth trade-off which is supercritical not only measured in variable size but also in formula size, and which has non-trivial robustness. More specifically, we prove that low width implies depth superlinear in the formula size, where the width is allowed to go up to twice the minimum.

We give analogous trade-offs for the Weisfeiler–Leman algorithm, a fundamental tool in graph isomorphism testing. Namely, for all $k \geq 2$ and $c \leq k-2$, if N is large enough, we show that there are vertex-size N graph pairs that are distinguishable by k-dimensional Weisfeiler–Leman, but even with dimension $k + c$ the algorithm nonetheless requires $\Omega(N^{k/(c+2)})$ many iterations. This improves the result in [7] which was proved in the case $c = 0$, solving an open problem there asking for lower bounds that hold for dimensions larger than the minimum. The result also translates into trade-offs between number of variables and quantifier depth in first-order logic.

Both results follow from lower bounds on a combinatorial game, closely linked to Tseitin formulas and Cai-Fürer-Immerman graphs. The game is the *compressed Cop-Robber game* introduced by [7]. It is a variant of the classical Cop-Robber game on a graph, where in addition a vertex-identification and edge-identification is posed, called a "compression". From a proof-complexity perspective, the compression induces a structured variable substitution under which the (Tseitin) formula size shrinks, a feature that is not possessed by the popular variable substitution based on XOR gadgets and expander graphs.

Our main technical contribution is a new compression scheme of the game and its analysis. Namely, for each $c \in \{1, \ldots, k-1\}$ we give a compressed game where $k + 1$ Cops can win, but the Robber can survive $\Omega(n^{k/(c+1)})$ rounds even against $k + c$ Cops.

References

[1] Paul Beame, Chris Beck, and Russell Impagliazzo, *Time-Space Tradeoffs in Resolution: Superpolynomial Lower Bounds for Superlinear Space*, SIAM Journal on Computing, 2016, vol. 45, no. 4, 1612–1645. Preliminary version in *STOC '12*.

[2] Christoph Berkholz, *On the Complexity of Finding Narrow Proofs*, Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science (FOCS '12), 2012, 351–360.

[3] Christoph Berkholz and Jakob Nordström, *Supercritical Space-Width Trade-offs for Resolution*, SIAM Journal on Computing, 2020, vol. 49, no. 1, 98–118. Preliminary version in *ICALP '16*.

[4] Chris Beck, Jakob Nordström, and Bangsheng Tang, *Some Trade-off Results for Polynomial Calculus*, Proceedings of the 45th Annual ACM Symposium on Theory of Computing (STOC '13), 2013, 813–822.

[5] Sam Buss and Neil Thapen, *A Simple Supercritical Tradeoff between Size and Height in Resolution*, 2024, Technical Report, Electronic Colloquium on Computational Complexity (ECCC), TR24-001.

[6] Noah Fleming, Toniann Pitassi, and Robert Robere, *Extremely deep proofs*, Proceedings of the 13th Innovations in Theoretical Computer Science Conference (ITCS '22), 2022, 70:1–70:23.

[7] Martin Grohe, Moritz Lichter, Daniel Neuen, and Pascal Schweitzer, *Compressing CFI Graphs and Lower Bounds for the Weisfeiler-Leman Refinements*, Proceedings of the 64th IEEE Annual Symposium on Foundations of Computer Science (FOCS '23), 2023, 798–809.

[8] Alexander Razborov, *A New Kind of Tradeoffs in Propositional Proof Complexity*, Journal of the ACM, 2016, vol. 63, no. 2, 16:1–16:14.

Proof Complexity and QBF

Meena Mahajan

This talk gave a short overview of proof complexity for *Quantified Boolean Formulas* (QBFs).

While traditionally the complexity of propositional proofs has been at the centre of research, the past two decades have witnessed a surge in proof complexity of QBFs. Some of the main paradigms used to extend propositional proof systems to QBFs include expansion, universal reduction, literal merging, and incremental strategy construction. (In particular, applying these paradigms gives multiple QBF proof systems based on resolution alone.) Soundness is often demonstrated by proving that from proofs in these systems, Herbrand functions (equivalently, winning strategies for the universal player in the two-player evaluation game) can be extracted.

There are not too many techniques for proving lower bounds in QBF proof systems. In most systems, propositional hardness transfers directly. But this is not the "genuine" QBF hardness we seek to understand, the hardness that would persist in a QBF solver even given access to, say, a SAT oracle. The principal technique for understanding such hardness is transferring computational hardness. Proofs contain, even if implicitly, information about winning strategies. Identifying the right computational model in which such extracted strategies can be computed enables the required transfer.

The most successful practical SAT solvers are based on the CDCL (Conflict-Driven Clause Learning) template, which is known to be equivalent to Resolution. In the QBF world, not only is there no unique analogue of Resolution, there is also no unique way of extending the algorithm template to Quantified QCDCL. Recent work has formalised proof systems underlying QCDCL-style algorithms and has also proposed more generalised proof systems, providing some analagues of the CDCL=Resolution equivalence.

An overview of QBF proof complexity can be found in [1], and of relations between QBF solving and proof complexity in [2].

References

[1] Olaf Beyersdorff. Proof complexity of quantified boolean logic – a survey. In Marco Benini, Olaf Beyersdorff, Michael Rathjen, and Peter Schuster, editors, *Mathematics for Computation (M4C)*, chapter Chapter 15, pages 397–440. World Scientific, 2023.

[2] Olaf Beyersdorff, Mikolás Janota, Florian Lonsing, and Martina Seidl. Quantified boolean formulas. In Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors, *Handbook of Satisfiability - Second Edition*, volume 336 of *Frontiers in Artificial Intelligence and Applications*, pages 1177–1221. IOS Press, 2021.

Dependency Schemes in CDCL-Based QBF Solving: A Proof Theoretic Study

Abhimanyu Choudhury

(joint work with Meena Mahajan)

With the success of propositional SAT solvers, there are many ambitious attempts now to tackle more expressive/succinct formalisms. In particular, for the PSPACE-complete problem of deciding the truth of Quantified Boolean Formulas (QBF), there are now many solvers, as well as a rich (and still growing) theory about the underlying formal proof systems. Designing solvers for QBFs is a useful enterprise because many industrial applications seem to lend themselves more naturally to expressions involving both existential and universal quantifiers; see for instance [1,2]. The proof system Resolution can be lifted to the QBF setting in many ways. The "`CDCL` way" is to add a universal reduction rule, giving rise to the system `Q-Res` and the more general `QU-Res`. Allowing contradictory literals to be merged under certain conditions gives rise to the system long-distance Q-Resolution `LDQ-Res`. Another "`CDCL`" way is to lift the `CDCL` algorithm itself to a QCDCL algorithm: decide values of variables, usually respecting the order of quantified alternation, propagate unit constraints, interpreting unit modulo universal reductions, repeat until a conflict is reached, learn a new clause, backtrack and continue. For false formulas, the learning process yields a long-distance Q-resolution refutation. In [3], a formal proof system `QCDCL` was abstracted out of the QCDCL algorithm.

A heuristic that has been found to be useful in many QBF solvers, and has been formalised in proof systems, is to eliminate easily-detectable spurious dependencies. In a prenex QBF, a variable "depends" on the variables preceding it in the quantifier prefix; where "depends" means that a Herbrand/Skolem function

for the variable is a function of the preceding variables. However, a Herbrand-function or countermodel may not really need to know the values of all preceding variables. A dependency scheme filters out as many of such unnecessary dependencies as it can detect, producing what is in effect a Dependency QBF, DQBF. Although DQBF is a significantly richer formalism that is known to be NEXP-complete (see [4, 5]), these heuristics are not aiming to solve DQBFs in general. Rather, they algorithmically detect spurious dependencies and disregard them as the algorithm proceeds.

Now, the universal reduction rule in the proof systems (say in Q-Res, LDQ-Res) can be applied in more settings because there are fewer dependencies, and this can shorten refutations significantly. See for instance [6–8]. Note that the use of a dependency scheme must be proven to be sound and complete, and this in itself is often quite involved. The notion of a dependency scheme being "normal" was introduced in [8], where it is shown that adding any normal dependency scheme to LDQ-Res preserves soundness and completeness.

We examine how the usage of a dependency scheme can affect proof systems underlying the QCDCL algorithm. Specifically, we focus on the proof system QCDCL , underlying most QCDCL-based solvers, and on the dependency scheme $\mathtt{D}^{\mathtt{rrs}}$ which has been studied in the context of Q-Res and LDQ-Res, see [6–8]. We note that the proof system QCDCL can be made aware of dependency schemes in more than one way. We identify two natural ways: (1) use a dependency scheme D to preprocess the formula, performing reductions in the initial clauses whenever permitted by the scheme, and (2) use a dependency scheme D in the QCDCL algorithm itself, in enabling unit propagations and in learning clauses. Denoting the first way as $\mathtt{D}+\mathtt{QCDCL}$ and the second as $\mathtt{QCDCL}(\mathtt{D})$, and noting that we could even use different dependency schemes in both these ways, we obtain the system $\mathtt{D}_1+\mathtt{QCDCL}(\mathtt{D}_2)$. When $\mathtt{D}_1$ and $\mathtt{D}_2$ are both the trivial dependency scheme $mathttD^{trv}$ inherited from the linear order of the quantifier prefix, this system is exactly QCDCL.

Our contributions are as follows:

(1) We formalise the proof system $\mathtt{D}'+\mathtt{QCDCL}(\mathtt{D})$ for dependency schemes $\mathtt{D},\mathtt{D}'$, and note that whenever $\mathtt{D}',\mathtt{D}$ are normal schemes, $\mathtt{D}'+\mathtt{QCDCL}(\mathtt{D})$ is sound and complete .
(2) For $\mathtt{D},\mathtt{D}'\in\{\mathtt{D}^{\mathtt{trv}},\mathtt{D}^{\mathtt{rrs}}\}$, we study the four systems $\mathtt{D}'+\mathtt{QCDCL}(\mathtt{D})$. As observed above, one of them is QCDCL itself, while the others are new systems. We compare these systems with each other and show that they are all pairwise incomparable We also show that each of them is incomparable with each of the systems $\mathtt{QCDCL}^{\mathtt{LEV-ORD}}_{\mathtt{NO-RED}}$ Q-Res, $\mathtt{Q}(\mathtt{D}^{\mathtt{rrs}})$Res and QU-Res.

In other words, making QCDCL algorithms dependency-aware is a "mixed bag": in some situations this shortens runs while in others it is disadvantageous.

References

[1] Shukla, A., Biere, A., Pulina, L. and Seidl, M., 2019, November. A survey on applications of quantified Boolean formulas. In 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI) (pp. 78–84). IEEE.

[2] Beyersdorff, O., Janota, M., Lonsing, F. and Seidl, M., 2021. Quantified boolean formulas. In Handbook of Satisfiability (pp. 1177–1221). IOS Press.

[3] Beyersdorff, O. and Böhm, B., 2023. Understanding the relative strength of QBF CDCL solvers and QBF resolution. Logical Methods in Computer Science, 19.

[4] Peterson, G., Reif, J. and Azhar, S., 2001. Lower bounds for multiplayer noncooperative games of incomplete information. Computers & Mathematics with Applications, 41(7-8), pp.957–992.

[5] Scholl, C. and Wimmer, R., 2018, June. Dependency quantified Boolean formulas: An overview of solution methods and applications. In International Conference on Theory and Applications of Satisfiability Testing (pp. 3–16). Cham: Springer International Publishing.

[6] Slivovsky, F. and Szeider, S., 2016. Soundness of Q-resolution with dependency schemes. Theoretical Computer Science, 612, pp.83–101

[7] Blinkhorn, J. and Beyersdorff, O., 2017. Shortening QBF proofs with dependency schemes. In Theory and Applications of Satisfiability Testing–SAT 2017: 20th International Conference, Melbourne, VIC, Australia, August 28–September 1, 2017, Proceedings 20 (pp. 263–280). Springer International Publishing.

[8] Peitl, T., Slivovsky, F. and Szeider, S., 2019. Long-distance Q-resolution with dependency schemes. Journal of Automated Reasoning, 63, pp.127–155.

Polynomial Calculus for Quantified Boolean Logic: Circuit Characterisation and Lower Bounds

Kaspar Kasche

(joint work with Olaf Beyersdorff, Luc Nicolas Spachmann)

We research the extension of the proof system PC (Polynomial Calculus) from propositional logic to Quantified Boolean Formulas. This extension is called $\mathcal{Q}$-PC and is defined by a general construction in [2].

Our first result is a tight circuit characterisation of $\mathcal{Q}$-PC proof size by circuit size in an appropriate circuit model. The circuit model in question is a generalisation of decision lists [7], which are lists of simple statements of the form: If *(condition on existential variables)* Then *(assignment to universal variables)*. The decision lists – termed PDLs here for *p*olynomial *d*ecision *l*ists – have polynomial equations in existential variables as conditions and compute a complete assignment to the universal variables. Semantically, a PDL for a quantified set of polynomial equations Φ computes a countermodel for Φ in the two-player game semantics of QBFs.

We show that the minimal proof size for Φ (of bounded quantifier complexity) in $\mathcal{Q}$-PC is polynomially equivalent to the minimal size of a PDL for Φ. In fact, we show a more general result that applies to a whole class of QBF proof systems with bounded capacity [6] (and fulfilling some closure properties). The result is parameterised by the lines of the proof system, which in turn correspond to the conditions in the decision lists. This generalises a result for Q-Resolution [4] and lifts it to $\mathcal{Q}$-PC.

Having the PDL characterisation in place, we can obtain a size-degree result, relating minimal proof size in $\mathcal{Q}$-PC to the minimal degree of polynomials in the refutation. This is similar in spirit to the size-degree method known for propositional PC [5], albeit the actual relation is different and includes the quantifier depth of the QBF. Technically, the result is shown via the degree-preserving transfer from $\mathcal{Q}$-PC to PDLs and back explained above, together with an additional size-degree relation that we show for PDLs. The technique is similar to a prior size-width result for Q-Resolution [4].

Having both the PDL characterisation and size-degree relation at hand opens the door to new lower bounds for degree and size in $\mathcal{Q}$-PC. Specifically, we show that the parity and more generally the modulo k functions mod_n^k on n variables as well as the majority function maj_n all require high-degree PDLs over all subfields of $\mathbb{C}$. Using a general construction from [1, 2] we can turn any Boolean function f into a QBF $\mathcal{Q}$-f that has f as its only countermodel. Together with our results above this implies that the $\mathcal{Q}$-mod_n^k and $\mathcal{Q}$-maj_n QBFs require both linear degree and exponential monomial size in $\mathcal{Q}$-PC.

In addition to using the size-degree method to prove lower bounds for PDLs and hence for $\mathcal{Q}$-PC proofs, we show that for finite fields of characteristic p, PDLs can be efficiently transformed into $\mathsf{AC}^0[p]$ circuits. This allows to directly transfer circuit lower bounds of [3] into $\mathcal{Q}$-PC proof lower bounds. As a result, either if F and G are both finite fields of different characteristics, or if F is finite and G is a subfield of $\mathbb{C}$, then the systems $\mathcal{Q}$-PC over F and G are incomparable.

References

[1] Beyersdorff, O., Chew, L. & Janota, M. New Resolution-Based QBF Calculi and Their Proof Complexity. (2019)

[2] Beyersdorff, O., Bonacina, I., Chew, L. & Pich, J. Frege Systems for Quantified Boolean Logic. *J. ACM.* **67**, 9:1-9:36 (2020)

[3] Smolensky, R. Algebraic methods in the theory of lower bounds for Boolean circuit complexity. *(STOC)*. pp. 77-82 (1987)

[4] Beyersdorff, O., Blinkhorn, J., Mahajan, M. & Peitl, T. Hardness Characterisations and Size-width Lower Bounds for QBF Resolution. *ACM Trans. Comput. Log.*. **24**, 10:1-10:30 (2023)

[5] Impagliazzo, R., Pudlák, P. & Sgall, J. Lower Bounds for the Polynomial Calculus and the Gröbner Basis Algorithm. *Comput. Complex.*. **8**, 127-144 (1999)

[6] Beyersdorff, O., Blinkhorn, J. & Hinde, L. Size, Cost, and Capacity: A Semantic Technique for Hard Random QBFs. *Logical Methods In Computer Science*. **15** (2019)

[7] Rivest, R. Learning Decision Lists. *Machine Learning*. **2**, 229-246 (1987)

Proof Systems for MaxSAT

Ilario Bonacina

(joint work with Maria Luisa Bonet and Jordi Levy)

MaxSAT is the problem of finding an assignment satisfying the maximum number of clauses in a CNF formula. Proof systems for MaxSAT are formal systems that can be used to certify the optimum value of a MaxSAT instance. The interest

in studying them arises naturally from practical concerns related to SAT- and MaxSAT-solvers, although some of the systems for MaxSAT are also studied for theoretical motivations, for instance due to connections with TFNP classes, see for instance [6]. One relevant proof system for MaxSAT is *MaxSAT-Resolution* [1, 7] which proves a lower bound s on the minimum number of falsified clauses in a set of clauses F in the following way: The proof is a sequence of multisets of clauses $\Gamma_0, \ldots, \Gamma_\ell$ such that (1) $\Gamma_0 = F$, (2) Γ_ℓ contains at least s copies of the empty clause $\bot$, and (3) Γ_{i+1} is obtained from Γ_i one of the following two substitution rules:

$$\frac{C \vee x \quad C \vee \neg x}{C} \qquad\qquad \frac{C}{C \vee x \quad C \vee \neg x},$$

where C are clauses and x is a variable. The clauses in a MaxSAT-Resolution proof can be thought as being *weighted clauses* with weight 1.

In this talk we survey some of the consequences of considering the natural generalization of MaxSAT-Resolution to weighted clauses using weights in $\{\pm 1\}$, or natural/integer weights encoded in binary. When negative weights are allowed the soundness of the systems is not automatically enforced and extra conditions on Γ_ℓ are needed. Varying those extra conditions and the weights allowed in the proofs, we have systems of different strength. For instance, (semi-)algebraic static proof systems such as Nullstellensatz (over $\mathbb{Z}$) and Sherali-Adams can be described naturally in this language [3, 5]. This can be used, for instance, to show how natural combinatorial principles capture the strength of Nullstellensatz (over $\mathbb{Z}$) and Sherali-Adams, with unary and binary coefficients [4].

In [2] we showed a similar approach to construct proof systems for MaxSAT based not on weighted clauses but on weighted polynomials. Starting from Polynomial Calculus over a finite field we showed how to adapt its inference rules to have a sound and complete system for MaxSAT. As in the case of weighted clauses, restricting the weights allowed also restricts the strength of the resulting MaxSAT system. In presence of integer weights, Polynomial Calculus for MaxSAT gives a natural proof system strictly stronger than Sherali-Adams while allowing the modular reasoning enabled by extending Polynomial Calculus over $\mathbb{F}_q$.

References

[1] M. L. Bonet, J. Levy, F. Manyà, *Resolution for Max-SAT*, Artif. Intell. (2007) **171**, 606–618.

[2] I. Bonacina, M. L. Bonet, J. Levy, *Polynomial Calculus for MaxSAT*, In: Theory and Applications of Satisfiability Testing (SAT 2023), pp. 5:1–5:18 (2023)

[3] I. Bonacina, M. L. Bonet, J. Levy, *Weighted, circular and semi-algebraic proofs*, JAIR (2024) **79**, 447–482

[4] I. Bonacina, M. L. Bonet, *On the strength of Shertali-Adams and Nullstellensatz as propositional proof systems*, In: Symposium on Logic in Computer Science (LICS 2022), pp. 25:1–25:12 (2022)

[5] M. L. Bonet, J. Levy, *Equivalence Between Systems Stronger Than Resolution*, In: Theory and Applications of Satisfiability Testing (SAT 2020), pp. 166–181 (2020)

[6] M. Göös, A. Hollender, S. Jain, G. Maystre, W. Pires, R. Robere, R. Tao, *Separations in Proof Complexity and TFNP*, In: Foundations of Computer Science (FOCS 2022), pp. 1150–1161 (2022)
[7] J. Larrosa, F. Heras, *Resolution in Max-SAT and its relation to local consistency in weighted CSPs*, In: Int. Joint Conf. on Artificial Intelligence (IJCAI 2005), pp. 193–198 (2005)

Certifying Combinatorial Solving Using Cutting Planes with Strengthening Rules

Jakob Nordström

Combinatorial optimization is the use of mathematical techniques to solve problems where, loosely speaking, solutions have to be constructed by combining objects in suitable ways, but where these objects cannot be subdivided (i.e., they are discrete). This leads to a combinatorial explosion in the number of cases that need to be considered, and in theory many combinatorial problems are known to be computationally very challenging (typically **NP**-hard or hard for even stronger complexity classes).

On the applied side, however, the last couple of decades has witnessed a revolution in combinatorial optimization algorithms in paradigms such as Boolean satisfiability (SAT) solving and optimization, constraint programming, and mixed integer linear programming, with modern so-called combinatorial solvers being used routinely to solve large-scale real-world problems. There is currently a very limited understanding of why these algorithms are so much more successful than theory would predict, and this is becoming more and more of an obstacle to further progress. Another critical concern is that modern solvers struggle with correctness. It is well documented in the literature that state-of-the-art solvers, both in academia and industry, sometimes erroneously claim optimality, or return "solutions" that do not satisfy the constraints in the input, or even claim simultaneously a solution and a bound ruling out this very solution. This can be fatal for applications where correctness is a non-negotiable demand.

The purpose of this presentation was to describe how proof complexity can be leveraged to design *certifying solvers*, which output not only an answer but also a machine-verifiable proof that this answer is correct—this process is also known as *proof logging*. With such a certifying combinatorial solver, the workflow becomes as follows (see also Figure 1):

(1) Run the solver on a problem to obtain not only a result, but also a proof.
(2) Feed the problem, result, and proof to a special computer program, called a *proof checker*.
(3) Accept the result if the proof checker says that the proof is valid.

Importantly, such a proof should be possible to produce with minimal overhead by the solver, which means that the proof system used has to be very expressive. At the same time, one would want the proof to be based on very simple rules, so that it is obvious how to verify mechanically that it is correct. Also, reading and understanding the proof should not require knowing the inner workings of the

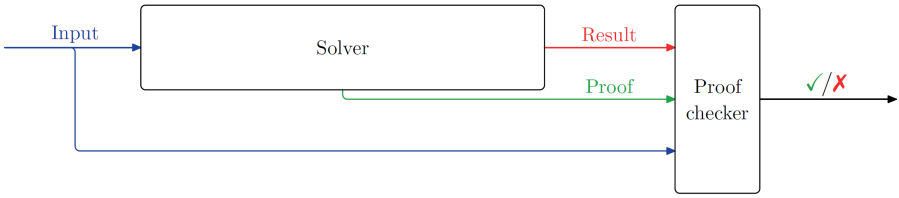

FIGURE 1. Schematic workflow for solver with proof logging.

solver or trusting that it has been implemented correctly, but should be possible with a fully independent, stand-alone proof checker.

Asking for all of this is quite a tall order, and it is far from clear a priori whether one should expect this to be possible in practice. Quite surprisingly, it turns out that the cutting planes proof system [4], if suitably extended with so-called strengthening rules, seems to hit a sweet spot between simplicity and expressivity, making it a convenient proof system for a wide range of applications. A proof checker for this proof system has been implemented in the tool VERIPB [18]. To date, VERIPB has been used to develop certifying versions of state-of-the-art solvers and techniques in

- Boolean satisfiability (SAT) solving (including advanced techniques such as Gaussian elimination [12] and symmetry breaking [3]);
- SAT-based optimization (MaxSAT solving) [1, 2, 17] including preprocessing [14];
- SAT-based pseudo-Boolean solving [11];
- subgraph solving (maximum clique, subgraph isomorphism, and maximum common connected subgraph) [7–9];
- dynamic programming and decision diagrams [5];
- presolving in 0–1 integer linear programming [13]; and
- constraint programming [6, 10, 15, 16].

The proof system underlying VERIPB has been developed in a sequence of joint works with Bart Bogaerts, Stephan Gocht, Ciaran McCreesh, Magnus O. Myreen, Andy Oertel, and Yong Kiam Tan.

REFERENCES

[1] Jeremias Berg, Bart Bogaerts, Jakob Nordström, Andy Oertel, and Dieter Vandesande. Certified core-guided MaxSAT solving. In *Proceedings of the 29th International Conference on Automated Deduction (CADE-29)*, volume 14132 of *Lecture Notes in Computer Science*, pages 1–22. Springer, July 2023.

[2] Jeremias Berg, Bart Bogaerts, Jakob Nordström, Andy Oertel, Tobias Paxian, and Dieter Vandesande. Certifying without loss of generality reasoning in solution-improving maximum satisfiability. In *Proceedings of the 30th International Conference on Principles and Practice of Constraint Programming (CP '24)*, September 2024. To appear.

[3] Bart Bogaerts, Stephan Gocht, Ciaran McCreesh, and Jakob Nordström. Certified dominance and symmetry breaking for combinatorial optimisation. *Journal of Artificial Intelligence Research*, 77:1539–1589, August 2023. Preliminary version in *AAAI '22*.

[4] William Cook, Collette Rene Coullard, and György Turán. On the complexity of cutting-plane proofs. *Discrete Applied Mathematics*, 18(1):25–38, November 1987.

[5] Emir Demirović, Ciaran McCreesh, Matthew McIlree, Jakob Nordström, Andy Oertel, and Konstantin Sidorov. Pseudo-Boolean reasoning about states and transitions to certify dynamic programming and decision diagram algorithms. In *Proceedings of the 30th International Conference on Principles and Practice of Constraint Programming (CP '24)*, September 2024. To appear.

[6] Jan Elffers, Stephan Gocht, Ciaran McCreesh, and Jakob Nordström. Justifying all differences using pseudo-Boolean reasoning. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI '20)*, pages 1486–1494, February 2020.

[7] Stephan Gocht, Ross McBride, Ciaran McCreesh, Jakob Nordström, Patrick Prosser, and James Trimble. Certifying solvers for clique and maximum common (connected) subgraph problems. In *Proceedings of the 26th International Conference on Principles and Practice of Constraint Programming (CP '20)*, volume 12333 of *Lecture Notes in Computer Science*, pages 338–357. Springer, September 2020.

[8] Stephan Gocht, Ciaran McCreesh, Magnus O. Myreen, Jakob Nordström, Andy Oertel, and Yong Kiam Tan. End-to-end verification for subgraph solving. In *Proceedings of the 368h AAAI Conference on Artificial Intelligence (AAAI '24)*, pages 8038–8047, February 2024.

[9] Stephan Gocht, Ciaran McCreesh, and Jakob Nordström. Subgraph isomorphism meets cutting planes: Solving with certified solutions. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI '20)*, pages 1134–1140, July 2020.

[10] Stephan Gocht, Ciaran McCreesh, and Jakob Nordström. An auditable constraint programming solver. In *Proceedings of the 28th International Conference on Principles and Practice of Constraint Programming (CP '22)*, volume 235 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 25:1–25:18, August 2022.

[11] Stephan Gocht, Ruben Martins, Jakob Nordström, and Andy Oertel. Certified CNF translations for pseudo-Boolean solving. In *Proceedings of the 25th International Conference on Theory and Applications of Satisfiability Testing (SAT '22)*, volume 236 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 16:1–16:25, August 2022.

[12] Stephan Gocht and Jakob Nordström. Certifying parity reasoning efficiently using pseudo-Boolean proofs. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI '21)*, pages 3768–3777, February 2021.

[13] Alexander Hoen, Andy Oertel, Ambros Gleixner, and Jakob Nordström. Certifying MIP-based presolve reductions for 0–1 integer linear programs. In *Proceedings of the 21st International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research (CPAIOR '24)*, volume 14742 of *Lecture Notes in Computer Science*, pages 310–328. Springer, May 2024.

[14] Hannes Ihalainen, Andy Oertel, Yong Kiam Tan, Jeremias Berg, Matti Järvisalo, Magnus O. Myreen, and Jakob Nordström. Certified MaxSAT preprocessing. In *Proceedings of the 12th International Joint Conference on Automated Reasoning (IJCAR '24)*, volume 14739 of *Lecture Notes in Computer Science*, pages 396–418. Springer, July 2024.

[15] Matthew McIlree and Ciaran McCreesh. Proof logging for smart extensional constraints. In *Proceedings of the 29th International Conference on Principles and Practice of Constraint Programming (CP '23)*, volume 280 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 26:1–26:17, August 2023.

[16] Matthew McIlree, Ciaran McCreesh, and Jakob Nordström. Proof logging for the circuit constraint. In *Proceedings of the 21st International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research (CPAIOR '24)*, volume 14743 of *Lecture Notes in Computer Science*, pages 38–55. Springer, May 2024.

[17] Dieter Vandesande, Wolf De Wulf, and Bart Bogaerts. QMaxSATpb: A certified MaxSAT solver. In *Proceedings of the 16th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR '22)*, volume 13416 of *Lecture Notes in Computer Science*, pages 429–442. Springer, September 2022.
[18] VeriPB: Verifier for pseudo-Boolean proofs. `https://gitlab.com/MIAOresearch/software/VeriPB`.

Bounds on the Total Coefficient Size of Nullstellensatz Proofs of the Pigeonhole Principle

Aaron Potechin
(joint work with Aaron Zhang)

Given a system $\{p_i = 0 : i \in [m]\}$ of m polynomial equations, a Nullstellensatz proof of infeasibility is an equality of the form $1 = \sum_{i=1}^{m} p_i q_i$ for some polynomials $\{q_i = 0 : i \in [m]\}$. Hilbert's Nullstellensatz[1] says that the Nullstellensatz proof system is complete, i.e., a system of polynomial equations has no solutions over an algebraically closed field if and only if there is a Nullstellensatz proof of infeasibility. However, Hilbert's Nullstellensatz does not give any bounds on how large a Nullsellensatz proof must be in order to refute an infeasible system of polynomial equations.

Previously, most research on Nullstellensatz has analyzed the size and degree of Nullstellenstaz proofs. In this work, instead of investigating the size or degree of Nullstellensatz proofs, we investigate the total coefficient size of Nullstellensatz proofs, i.e., the sum of the magnitudes of the coefficients of the monomials in the proof. Our main reason for this is that total coefficient size is a reasonably natural measure which is relatively unexplored (though there has been considerable research on closely related measures such as unary Nullstellensatz size, unary Sherali-Adams size, and the total bit complexity of proofs [1–3, 5, 7]). That said, there are several other reasons why total coefficient size bounds are interesting.

First, analyzing the total coefficient size of proofs may give insight into proof size in settings where we currently cannot prove size lower bounds. If we can prove a large total coefficient size lower bound, this shows that any proof must either have large size or involve large coefficients. Unless there is a reason to suspect that large coefficients are helpful for making the proof shorter, this gives considerable evidence for a lower bound on proof size.

Second, lower bounds on total coefficient size have some direct implications. As observed by [5], a total coefficient size lower bound for the stronger Sherali-Adams proof system implies a lower bound for the reversible resolution proof system which captures the Max-SAT resolution proof system (see [4]) for Max SAT. Similarly, [5] observes that a total coefficient size lower bound for Nullstellensatz implies a lower

[1]Technically, this is the weak form of Hilbert's Nullstellensatz. Hilbert's Nullstellensatz actually says that given polynomials $p_1, \ldots, p_m$ and another polynomial p, if $p(x) = 0$ for all x such that $p_i(x) = 0$ for each $i \in [m]$ then there exists a natural number r such that p^r is in the ideal generated by $p_1, \ldots, p_m$.

bound for the reversible resolution with terminals proof system, which is a weaker variant of reversible resolution.

Finally, investigating the total coefficient size of proofs gives insight into the following question. Are there natural examples where having fractional coefficients greatly reduces the total coefficient size needed for Nullstellensatz and/or Sherali-Adams proofs? Proving total coefficient size lower bounds for a problem rules out this possibility for that problem. Conversely, if there is a natural example where the minimum proof size is large but the total coefficient size is small, this would be quite interesting.

In this work, we show that the minimum total coefficient size of a Nullstellensatz proof of the pigeonhole principle is $2^{\Theta(n)}$. More precisely, we show the following bounds.

Theorem 1. *For all $n \geq 2$, any Nullstellensatz proof of the pigeonhole principle with n pigeons and $n-1$ holes has total coefficient size $\Omega\left(n^{\frac{3}{4}}\left(\frac{2}{\sqrt{e}}\right)^n\right)$.*

We note that this lower bound also holds for the functional pigeonhole principle, where each pigeon must go to exactly one hole (instead of at least one hole).

Theorem 2. *For all $n \geq 2$, there is a Nullstellensatz proof of the pigeonhole principle with n pigeons and $n-1$ holes with total coefficient size at most 2^{5n}.*

This is joint work with Aaron Zhang which will appear in ICALP 2024. The full version of our paper is on arXiv [6]. This research was supported by NSF grant CCF-2008920 and NDSEG fellowship F-9422254702.

References

[1] Yaroslav Alekseev. A lower bound for polynomial calculus with extension rule. In *Proceedings of the 36th Computational Complexity Conference*, CCC '21, Dagstuhl, DEU, 2021. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[2] Yaroslav Alekseev, Dima Grigoriev, Edward A. Hirsch, and Iddo Tzameret. Semi-algebraic proofs, ips lower bounds, and the tau-conjecture: can a natural number be negative? In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2020, page 54–67, New York, NY, USA, 2020. Association for Computing Machinery.

[3] Ilario Bonacina and Maria Luisa Bonet. On the strength of sherali-adams and nullstellensatz as propositional proof systems. In *Proceedings of the 37th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '22, New York, NY, USA, 2022. Association for Computing Machinery.

[4] María Luisa Bonet, Jordi Levy, and Felip Manyà. Resolution for max-sat. *Artificial Intelligence*, 171(8):606–618, 2007.

[5] M. Goos, A. Hollender, S. Jain, G. Maystre, W. Pires, R. Robere, and R. Tao. Separations in proof complexity and tfnp. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1150–1161, Los Alamitos, CA, USA, nov 2022. IEEE Computer Society.

[6] Aaron Potechin and Aaron Zhang. Bounds on the total coefficient size of nullstellensatz proofs of the pigeonhole principle and the ordering principle, 2022.

[7] S. F. de Rezende, A. Potechin, and K. Risse. Clique is hard on average for unary Sherali-Adams. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 12–25, Los Alamitos, CA, USA, nov 2023. IEEE Computer Society.

Strength of the Dominance Rule

Neil Thapen
(joint work with Leszek Aleksander Kołodziejczyk)

It has become standard that, when a SAT solver decides that a CNF Γ is unsatisfiable, it produces a certificate of unsatisfiability in the form of a refutation of Γ in some proof system. The system typically used is DRAT, which is equivalent to extended resolution (ER) – for example, until this year DRAT refutations were required in the annual SAT competition.

Recently Bogaerts et al. [1] introduced a new proof system, associated with the tool VeriPB, which is at least as strong as DRAT and is further able to handle certain symmetry-breaking techniques. We show that this system simulates the proof system G_1, which allows limited reasoning with QBFs and forms the first level above ER in a natural hierarchy of proof systems [2]. This hierarchy is not known to be strict, but nevertheless this is evidence that the system of [1] is plausibly strictly stronger than ER and DRAT. In the other direction, we show that symmetry-breaking for a single symmetry can be handled inside ER.

References

[1] B. Bogaerts, S. Gocht, C. McCreesh and J. Nordström. *Certified dominance and symmetry breaking for combinatorial optimisation.* Journal of Artificial Intelligence Research, 77:1539–1589, 2023.

[2] J. Krajícek and P. Pudlák. *Quantified propositional calculi and fragments of bounded arithmetic.* Z. Math. Logik Grundlag. Math., 36(1):29–46, 1990.

Provability of Circuit Size Hierarchies

Marco Carmosino
(joint work with Valentine Kabanets, Antonina Kolokolova and Igor C. Oliveira)

A *gate* is an atomic device that computes a single Boolean function. A *circuit* is an arrangement of wires between gates. Each circuit computes a particular Boolean function on a fixed number of input bits by propagating values along the wires. To measure the circuit complexity of a function f, we fix a set of gates $\mathcal{B}$ — for example, two-bit $\{\mathsf{AND}, \mathsf{OR}\}$ and NOT gates — and count the minimum number of $\mathcal{B}$-gates required to compute f.

Despite decades of study, basic questions about circuit complexity remain open. Straightforward counting arguments show that most Boolean function on n bits require huge circuits: roughly $2^n/n$ gates [4]. Yet, explicit functions that require even super-linear circuit size are unknown. The best known lower bounds for circuits over $\mathcal{B}$ are $5n - o(n)$, but the explicit functions identified can be computed using only $5n + o(n)$ gates [1, 2]. So, new ideas are required for explicit circuit lower bounds in the general[1] setting.

[1]Super-polynomial lower bounds are indeed known for constant-depth circuit classes and under certain other structural restrictions.

Motivated by the apparent difficulty of making progress, some researchers are exploring mathematical logic to understand why some questions about circuits resist all known proof techniques. A weak fragment of Peano Arithmetic called PV_1 (for "Polynomially Verifiable") captures "efficient" reasoning, by limiting the induction principle to work only on formulas quantifying "small" numbers. Even so, PV_1 formalizes many theorems about computational complexity — including the Cook-Levin and PCP theorems — which seem to require intricate proofs [3].

The immediate meta-mathematical question is: how much circuit complexity can be accomplished inside PV_1? In particular, the Circuit Size Hierarchy (CSH) is a classical result: larger circuits compute strictly more functions. CSH is proved by straightforward counting and encoding of Boolean functions. But a "constructive" proof of CSH remains unknown, and the existing arguments do not produce explicit hard functions. Formally, we ask: is CSH a theorem of PV_1?

This talk shared preliminary evidence that it is difficult to prove CSH in PV_1. If CSH is a theorem of PV_1, then there are super-linear circuit lower bounds for a function computable in PTIME — a breakthrough. This reduces the problem of proving super-linear citcuit lower bounds to a question about the existence of feasible proofs.

However, our work remains in progress because it seems natural to ask for a better relationship between PV_1-provability of CSH and breakthrough circuit lower bounds. Suppose PV_1 proves that, for each $k \in \mathbb{N}$, circuits of size n^{5k} compute functions that are hard for circuits of size n^k. We hope to obtain from this assumption, for each k, a language $\mathcal{H}_k \in \mathsf{PTIME}$ that requires $n^{k+\varepsilon}$ circuit size, with $\varepsilon > 0$. Intuitively, despite our preliminary result, it remains open to "extract all the hardness" from a PV_1-proof of CSH.

References

[1] K. Amano and J. Tarui, *A well-mixed function with circuit complexity* $5n$*: Tightness of the Lachish-Raz-type bounds*, Theor. Comput. Sci., **412** no.18 (2011), 1646–1651.

[2] K. Iwama and H. Morizumi, *An Explicit Lower Bound of* $5n - o(n)$ *for Boolean Circuits*, MFCS Lecture Notes in Computer Science **2420** (2002), 353–364.

[3] J. Pich, *Logical strength of complexity theory and a formalization of the PCP theorem in bounded arithmetic*, Log. Methods Comput. Sci., **11** no. 2, (2015).

[4] C. E. Shannon, *The synthesis of two-terminal switching circuits*, Bell Systems Technical Journal **28** (1949), 59–98.

Reporter: Jonas Conneryd

Participants

Dr. Robert Andrews
Institute for Advanced Study
School of Mathematics
1 Einstein Drive
Princeton, NJ 08540
UNITED STATES

Prof. Dr. Albert Atserias
Dept. Ciències de la Computació
Universitat Politécnica de Catalunya
Omega-231
Jordi Girona Salgado, 1-3
08034 Barcelona
SPAIN

Prof. Dr. Paul Beame
Allen School of Computer Science
and Engineering
University of Washington
Box 352350
Seattle, WA 98195-2350
UNITED STATES

Prof. Dr. Arnold Beckmann
Department of Computer Science
Swansea University
Bay Campus
Swansea SA1 8EN
UNITED KINGDOM

Prof. Dr. Christoph Berkholz
Technische Universität Ilmenau
Postfach 10 05 65
98684 Ilmenau
GERMANY

Prof. Dr. Greg Blekherman
School of Mathematics
Georgia Institute of Technology
686 Cherry Street
Atlanta, GA 30332-0160
UNITED STATES

Dr. Ilario Bonacina
Universitat Politecnica de
Catalunya (UPC)
Jordi Girona 1-3
08034 Barcelona
SPAIN

Prof. Dr. Maria Luisa Bonet
Department of Computer Science
Universidad Politécnica de Catalunia
Jordi Girona Salgado, 1-3
08034 Barcelona
SPAIN

Prof. Dr. Samuel Buss
Department of Mathematics
University of California, San Diego
9500 Gilman Drive
La Jolla, CA 92093-0112
UNITED STATES

Dr. Igor Carboni Oliveira
Department of Computer Science
University of Warwick
Gibbet Hill Road
Coventry CV4 7AL
UNITED KINGDOM

Dr. Marco Carmosino
IBM Corporation
Thomas J. Watson Research Center
161 Washington St Apt 1105
Boston, MA 02135
UNITED STATES

Abhimanyu Choudhury
Institute of Mathematical Sciences
4th Cross Street, CIT Campus,
Tharamani
Tamil Nadu, Chennai 600 113
INDIA

Jonas Conneryd
Department of Computer Science
Lund University
Ole Römers Väg 3
22363 Lund
SWEDEN

Dr. Susanna F. de Rezende
Department of Computer Science
Lund University
221 00 Lund
SWEDEN

Dr. Joanna Fijalkow
CNRS & University of Bordeaux
351 Cours de la Libération
F-33405 Talence
FRANCE

Prof. Dr. Noah Fleming
Computer Science Department
Memorial University of Newfoundland
St. John's, NL A1C 5S7
CANADA

Prof. Dr. Nicola Galesi
Department of Computer, Control and
Management Engineering - "A. Ruberti"
Sapienza Università Roma
Via Ariosto 25
00185 Roma
ITALY

Yassine Ghannane
Department of Computer Science
University of Copenhagen
Universitetsparken 1
2100 København
DENMARK

Dr. Mika Göös
Ecole Polytechnique Fédérale de
Lausanne
EPFL IC THL5
Station 14
1015 Lausanne
SWITZERLAND

Prof. Dr. Martin Grohe
Lehrstuhl für Informatik VII
RWTH Aachen
Ahornstraße 55
52074 Aachen
GERMANY

Dr. Tuomas Hakoniemi
University of Helsinki
Department of Computer Science
Pietari Kalmin katu 5
P.O. Box PL 68
00014 Helsinki
FINLAND

Prof. Dr. Johan Håstad
Department of Mathematics
KTH Royal Institute of Technology
100 44 Stockholm
SWEDEN

Prof. Dr. Edward A. Hirsch
Department of Computer Science
Ariel University
Ariel 40700
ISRAEL

Dr. Pavel Hrubes
Institute of Mathematics of the AV CR
Žitná 25
115 67 Praha 1
CZECH REPUBLIC

Dr. Dmitry Itsykson
Department of Computer Science
Ben-Gurion University of the Negev
Beer-Sheva 84 105
ISRAEL

Duri Andrea Janett
Department of Computer Science
University of Copenhagen
Universitetsparken 1
2100 København
DENMARK

Dr. Emil Jeřábek
Institute of Mathematics of the Czech Academy of Sciences
Žitná 25
115 67 Praha 1
CZECH REPUBLIC

Kaspar Kasche
Institut für Informatik
Universität Jena
Ernst-Abbe-Platz 2
07743 Jena
GERMANY

Dr. Leszek Kołodziejczyk
Institute of Mathematics
University of Warsaw
ul. Banacha 2
02-097 Warszawa
POLAND

Dr. Antonina Kolokolova
Department of Computer Science
Memorial University of Newfoundland
St. John's, NL A1B 3X5
CANADA

Prof. Dr. Pravesh K. Kothari
Princeton University, and the Institute for Advanced Study
35 Olden Street
Princeton, NJ 08540-5233
UNITED STATES

Dr. Massimo Lauria
Dipartimento di Scienze Statistiche
Università Sapienza di Roma
P.le Aldo Moro 5
00185 Roma
ITALY

Prof. Dr. Meena Mahajan
The Institute of Mathematical Sciences
(a CI of Homi Bhabha National Institute HBNI)
CIT Campus, Taramani
Chennai, Tamil Nadu 600 113
INDIA

Prof. Dr. Moritz Müller
Universität Passau
Lehrstuhl Mathematische Logik
Dr.-Hans-Kapfinger Straße 30
94032 Passau
GERMANY

Prof. Dr. Jakob Nordström
Department of Computer Science
University of Copenhagen
Universitetsparken 1
2100 København
DENMARK

Dr. Shuo Pang
Department of Computer Science
University of Copenhagen
Universitetsparken 5
2100 København
DENMARK

Theodoros Papamakarios
Department of Mathematics
The University of Chicago
5734 South University Avenue
Chicago, IL 60637-1514
UNITED STATES

Prof. Dr. Toniann Pitassi
Columbia University
New York, NY 10027
UNITED STATES

Prof. Dr. Aaron Potechin
Department of Computer Science
The University of Chicago
5730 S. Ellis Avenue
Chicago, IL 60637-1514
UNITED STATES

Prof. Dr. Pavel Pudlák
Institute of Mathematics, Czech Academy of Sciences
Žitná 25
115 67 Praha 11567
CZECH REPUBLIC

Prof. Dr. Annie Raymond
Department of Mathematics and Statistics
University of Massachusetts, Amherst
710 N. Pleasant Street
01003-9305 Amherst Massachusetts
UNITED STATES

Prof. Dr. Alexander A. Razborov
Department of Mathematics and Computer Science
The University of Chicago
Ryerson Hall
1100 East 58th Street
Chicago, IL 60637
UNITED STATES

Dr. Kilian Risse
School of Computer and Communication Sciences
École Polytechnique Fédérale de Lausanne
1015 Lausanne
SWITZERLAND

Robert Robere
School of Computer Science
McGill University
3480 University Street
H3A0E9 Montréal
CANADA

Prof. Dr. Rahul Santhanam
Department of Computer Science
Oxford University
Wolfson Building
Parks Road
Oxford OX1 3QD
UNITED KINGDOM

Anastasia Sofronova
École Polytechnique Fédérale de Lausanne
1015 Lausanne
SWITZERLAND

Dmitry Sokolov
École Polytechnique Fédérale de Lausanne
1015 Lausanne
SWITZERLAND

Dr. Neil Thapen
Institute of Mathematics of the AV CR
Žitná 25
115 67 Praha 1
CZECH REPUBLIC

Dr. Jacobo Toran
Institut für Theoretische Informatik
Universität Ulm
Oberer Eselsberg
89069 Ulm
GERMANY

Dr. Madhur Tulsiani
Toyota Technological Institute at Chicago
6045 S Kenwood Av
Chicago, IL 60637
UNITED STATES

Dr. Iddo Tzameret
Department of Computing
Imperial College London
London SW7 2AZ
UNITED KINGDOM

Marc Vinyals
School of Computer Science
The University of Auckland
P.O. Box 92019
1142 Auckland
NEW ZEALAND