Report from Dagstuhl Seminar 12471

SAT Interactions

Edited by

Nadia Creignou[1], Nicola Galesi[2], Oliver Kullmann[3], and Heribert Vollmer[4]

1 **Université de Marseille, FR,** creignou@lif.univ-mrs.fr
2 **University of Rome "La Sapienza", IT,** galesi@di.uniroma1.it
3 **Swansea University, GB,** o.kullmann@swansea.ac.uk
4 **Leibniz Universität Hannover, DE,** vollmer@thi.uni-hannover.de

Abstract

This report documents the programme and outcomes of Dagstuhl Seminar 12471 "SAT Interactions". The seminar brought together researchers from different areas from theoretical computer science as well as the area of SAT solvers. A key objective of the seminar has been to initiate or consolidate discussions among the different groups for a fresh attack on one of the most important problems in theoretical computer science and mathematics.

Seminar 18.–23. November, 2012 – www.dagstuhl.de/12471

1998 ACM Subject Classification D.2.8 Complexity measures, F.2 Analysis of Algorithms and Problem Complexity, F.1.3 Complexity Measures and Classes, F.2.2 Nonnumerical Algorithms and Problems, F.4 Mathematical Logic and Formal Languages, G.2.1 Combinatorics

Keywords and phrases satisfiability problem, computational complexity, P-NP question, proof complexity, combinatorics, SAT-solvers, quantified Boolean formulas

Digital Object Identifier 10.4230/DagRep.2.11.87

Edited in cooperation with Johannes Ebbing

1 Executive Summary

Nadia Creignou
Heribert Vollmer

Brief Introduction to the Topic

Propositional satisfiability (or Boolean satisfiability) is the problem of determining whether the variables of a Boolean formula can be assigned truth values in such a way as to make the formula true. The satisfiability problem, SAT for short, stands at the crossroads of logic, graph theory, computer science, computer engineering and computational physics.

In particular SAT is of central importance in various areas of computer science including algorithmics, verification, planning and hardware design. It can express a wide range of combinatorial problems as well as many real-world ones. Due to its potential practical implications an intensive search has been done on how one could solve SAT problems in an automated fashion. The last decade has seen the development of practically-efficient algorithms for SAT, which can solve huge problems instances.

At the same time SAT is very significant from a theoretical point of view. Since the Cook-Levin's theorem, which has identified SAT as the first NP-complete problem, it has become a reference for an enormous variety of complexity statements. The most prominent one is the question "is P equal to NP?" Proving that SAT is not in P would answer this question negatively. Indeed, as stated by Richard Lipton on his blog *Gödel's Lost Letter and $P = NP$* (http://rjlipton.wordpress.com) such a proof matters since it would tell us why some computational problems are intrinsically more difficult than others, it would suggest new methods that would yield new insights on the fundamental nature of computation and it would help with goals of security for cryptographers.

During the past two decades, an impressive array of diverse techniques from mathematical fields, such as propositional logic, model theory, Boolean function theory, combinatorics, probability, and statistical physics has contributed to a better understanding of the SAT problem. Although significant progress has been made on several fronts, most of the central questions remain unsolved so far. One of the main aims of the Dagstuhl Seminar was to bring together researchers from different areas of activity in SAT (with an emphasize on mathematical aspects), so that they can communicate state-of-the-art advances and embark on a systematic interaction that will enhance the synergy between the different areas.

Organization of the Seminar and Activities

The workshop brought together 44 researchers from different areas of computer science and mathematics such as logic, complexity theory, algorithms, and proof complexity with complementary expertise. The participants consisted of both senior and junior researchers, including a number of postdocs and a few advanced graduate students.

Participants were invited to present their work and to communicate state-of-the-art advances. Twenty-five talks of various lengths took place over the five days of the workshop. Introductory and tutorial talks of 60 minutes were scheduled prior to workshop. Most of the remaining slots were filled, mostly with shorter talks, as the workshop commenced. The organizers considered it important to leave ample free time for discussion.

The tutorial talks were scheduled during the beginning of the week in order to establish a common background for the different communities that came together for the workshop. The presenters and topics were:

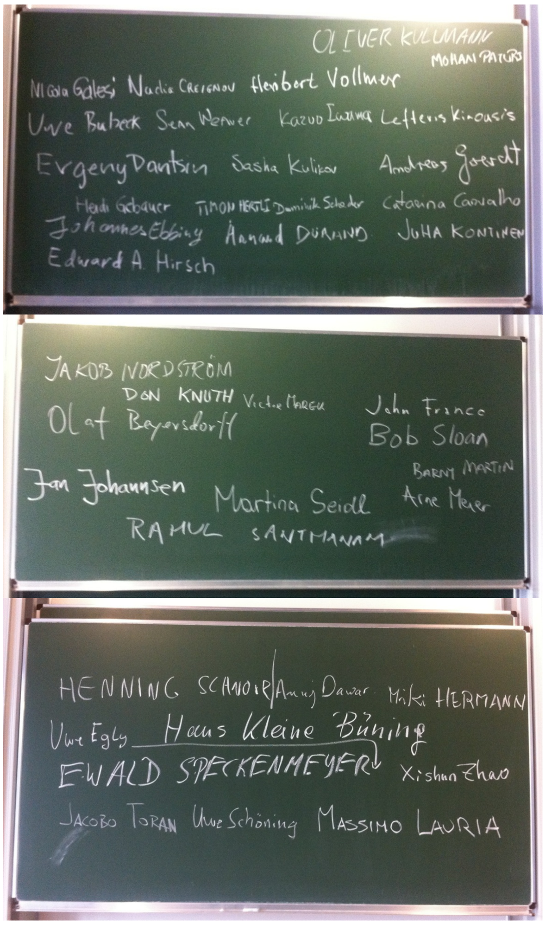
- Olaf Beyersdorff, Proof complexity
- Arne Meier, Complexity classifications for different satisfiability problems
- Victor Marek, Erdős' dream; SAT and combinatorics
- Uwe Bubeck, Quantified Boolean formulas: complexity and expressiveness
- Oliver Kullmann, The combinatorics of minimal unsatisfiability
- Martina Seidl, QBF solvers

Most of the tutorials were given by young researchers, reflecting the fact that the SAT community is dynamic and fast evolving.

A highlight of the seminar was the talk by Donald E. Knuth, delivered Wednesday morning, on "Satisfiability and the Art of Computer Programming". Knuth reported about his experiences while working on a chapter on satisfiability for the upcoming volume of his world-renowned series.

There were additionally 19 shorter talks. These talks covered a wide range of topics related to satisfiability. The different approaches discussed above in the seminar description were all very well represented by the different talks given during the five days of the seminar.

1. Combinatorics
   - Xishun Zhao, Finiteness conjecture on hitting minimal unsatisfiable formulas
   - Uwe Schöning, Probability distributions for local search and make versus break
   - Heidi Gebauer, Applications of (k, d)-trees



2. Complexity
   - Juha Kontinen, Dependence logic and complexity
   - Julian-Steffen Müller, A fragment of dependence logic characterizing PTIME
   - Alexander Kulikov, New lower and upper bounds for Boolean circuit complexity
   - Johannes Ebbing, Model checking for modal intuitionistic dependence logic
3. Proof complexity
   - Uwe Egly, Proof complexity for QBF
   - Jan Johannsen, Separating clause learning proof systems from (regular) resolution
   - Jakob Nordström, Relating proof complexity measures and practical hardness of SAT
   - Massimo Lauria, Open problems in proof complexity
4. Algorithms
   - Stefan Szeider, Fixed-parameter tractability and SAT
   - Mohan Paturi, Algorithmic expressivity and hardness of satisfiability
   - Dominik Scheder, Exponential lower bounds for the PPSZ k-SAT algorithm

5. Logic
   - Arnaud Durand, A criterion for tractability of counting solutions to uniform CSP
   - Hans Kleine Büning, On some configuration problems based on representations in propositional logic
6. Solvers
   - John Franco, Adding unsafe constraints to improve satisfiability performance
   - Sean Weaver, Satisfiability enhancements enabled by state machines

This classification is necessarily rough, as several talks crossed the boundaries between these areas, in keeping with the theme of the workshop. The broad scope of the talks extended even to areas not anticipated by the organizers, such as dependence logic. The workshop thus achieved its aim of bringing together researchers from various related communities to share state-of-the-art research.

Concluding Remarks and Future Plans

The organizers regard the workshop as a great success. Bringing together researchers from different areas of theoretical computer science fostered valuable interactions and led to fruitful discussions. Feedback from the participants was very positive as well. Many attendants expressed their wish for a continuation and stated that this seminar was among the most fruitful Dagstuhl seminars they attended.

Finally, the organizers wish to express their gratitude toward the Scientific Directorate of the Center for its support of this workshop, and hope to establish a series of workshops on *SAT Interactions* in the future.

2 Table of Contents

3 Overview of Talks

3.1 Proof Complexity

Olaf Beyersdorff (University of Leeds, GB)

This talk surveys important results from the area of propositional proof complexity. In the talk I will highlight motivations and applications of proof complexity and important techniques which have been developed to show lower bounds. In particular, I will explain a game-theoretic technique which characterises tree-like Resolution size and illustrate this technique by proving the optimal lower bound for the pigeonhole principle in tree-like Resolution.

3.2 Quantified Boolean Formulas: Complexity and Expressiveness

Uwe Bubeck (Universität Paderborn, DE)

Joint work of Bubeck, Uwe; Kleine Büning, Hans
Main reference U. Bubeck, "Model-Based Transformations for Quantified Boolean Formulas," in: Dissertations in Artificial Intelligence (DISKI), Vol. 329, Wolfgang Bibel (Ed.), IOS Press, ISBN 978-1-60750-545-7, 2010.
URL http://www.ub-net.de/bubeck-qbf-transformations-2010.pdf

We consider quantified Boolean formulas with free variables (QBF*) as an elegant way to represent Boolean functions. In the talk, we give an overview of fundamental concepts and complexity results, and we present suitable encoding techniques to compress propositional formulas by applying quantification.

We also relate QBF* to other representations of Boolean functions. In particular, we discuss the close relationship between existential quantification and Boolean circuits with unbounded fan-out [1, 2], as well as transformations between quantified Boolean formulas and nested Boolean functions (NBF) in both directions [3].

References

1 S. Aanderaa and E. Börger. *The Horn Complexity of Boolean Functions and Cook's Problem*. Proc. 5th Scandinavian Logic Symposium 1979, Aalborg University Press, 1979

2 H. Kleine Büning, X. Zhao, and U. Bubeck. *Resolution and Expressiveness of Subclasses of Quantified Boolean Formulas and Circuits*. Proc. 12th Intl. Conf. on Theory and Applications of Satisfiability Testing (SAT 2009), Springer, 2009

3 U. Bubeck and H. Kleine Büning. *Encoding Nested Boolean Functions as Quantified Boolean Formulas*. Journal on Satisfiability, Boolean Modeling and Computation (JSAT) 8(1): 101-116, 2012

3.3 Structural Tractability of Counting of Solutions to Conjunctive Queries

Arnaud Durand (University Paris-Diderot, FR)

Joint work of Durand, Arnaud; Mengel, Stefan

This talk survey some recent characterization results obtained on the counting complexity of subclasses of conjunctive queries (i.e. constraint satisfaction problem with projection). We prove that for counting of acyclic conjunctive queries (and many more fragments) it is possible to chart the tractability frontier. One of the main ingredients of this characterization is a new parameter associated to formulas that measure how free variables are spread into formulas.

3.4 Model Checking for Modal Intuitionistic Dependence Logic

Johannes Ebbing (Leibniz Universität Hannover, DE)

Joint work of Ebbing, Johannes; Lohmann, Peter; Yang, Fan

Modal intuitionistic dependence logic (MID) incorporates the notion of "dependence" between propositions into the usual modal logic and has connectives which correspond to intuitionistic connectives in a certain sense. It is the modal version of a variant of first-order dependence logic introduced by Väänänen in [1] considered by Abramsky and Väänänen [2] basing on Hodges' team semantics (1997). In this talk we give an overview on the computational complexity of the model checking problem for MID and its fragments built by restricting the operators allowed in the logics. In particular, we see that the model checking problem for MID is in general PSPACE-complete and that for propositional intuitionistic logic is coNP-complete.

References

1 J. Väänänen, *Dependence logic: A new approach to independence friendly logic*, London Mathematical Society student texts, no. 70, Cambridge University Press, 2007.
2 S. Abramsky and J. Väänänen, *From IF to BI*, Synthese 167 (2009), no. 2, 207–230.

3.5 Adding Unsafe Constraints to Improve the Performance of SAT Algorithms

John Franco (University of Cincinnati, US)

Joint work of Franco, John; Kouril, Michal
Main reference M. Kouril, J. Franco, "Resolution Tunnels for Improver SAT Solver Performance," in Proc. of the 8th Int'l Conf. on Theory and Applications of Satisfiability Testing (SAT'05), LNCS, Vol. 3569, pp, 143–158, 2005.
URL http://dx.doi.org/10.1007/11499107_11

For many families of SAT formulas the difficulty in solving an instance escalates exponentially with increasing instance size. A possible reason for this is that inferred constraints that reduce search space significantly are learned too late in the search to be effective. One attempt to control this is to add safe, uninferred constraints that are obtained from an analysis of the problem or the structure of the formula: for example symmetry breaking constraints. This approach proves effective in some but not all cases. We propose an alternative approach which is to add unsafe, uninferred constraints early on to reduce search space breadth at shallow depth and then retract those constraints when the search breadth is still small and will not get much bigger as search continues. By "unsafe constraint" we mean a constraint that may eliminate one or more satisfying assignments – hence there is a risk that all assignments of satisfiable instance may be eliminated.

We show, for example that in the case of formulas for solving van der Waerden number W(2,6), adding unsafe constraints produces a bound that turns out to be W(2,6). Knowledge of this bound and the conjecture that it was W(2,6) was eventually used by Kouril to custom design a solver that could prove definitively the value of W(2,6). Notable is the fact that the unsafe constraints are obtained from an analysis of solutions to smaller instances of the van der Waerden family and not from an analysis of the structure of the formulas or problem properties.

3.6 On Configuration Problems Based on Representations in Propositional Logic

Hans Kleine Büning (Universität Paderborn, DE)

Joint work of Bubeck, Uwe; Kleine Büning, Hans; Yan, Yuhan; Zhao, Xishun

We consider configuration problems, where the components are represented by propositional formulas. Configuration is the process of composing a system from a predefined set of components, while observing a set of given constraints and customer demands.

We focus on the computational complexity of the configuration problem for various sub-classes of formulas and architectures of the desired system. For example, the desired architecture can be a set of components or a circuit whose nodes are components computing Boolean functions

Moreover, we investigate the so-called specification problem in which we want to learn an unknown component given a partial solution of the configuration problem.

3.7 New Lower and Upper Bounds for Boolean Circuit Complexity

Alexander S. Kulikov (Steklov Inst. – St. Petersburg, RU)

Joint work of Demenkov, Evgeny; Kojevnikov, Arist; Kulikov, Alexander S.; Melanich, Olga; Mihajlin, Ivan; Yaroslavtsev, Grigory

In the first part of the talk, we will show how SAT-solvers can help to prove stronger upper bounds on the boolean circuit complexity. Roughly, the main idea is that circuits for some functions are naturally built from blocks of constant size. E.g., the well-known circuit that computes the binary representation of the sum of n input bits is built from n full adders and has size $5n$. One can then state the question "whether there exist a block of smaller size computing the same function" in terms of CNF- SAT and then ask SAT-solvers to verify this. Using this simple approach we managed to improve the upper bound for the above mentioned function to $4.5n$. This, in particular, implies that any symmetric function has circuit size at most $4.5n + o(n)$. We will also present improved upper bounds for some other symmetric functions.

In the second part we will present much simpler proofs of currently best known lower bounds on boolean circuit complexity. These are $3n - o(n)$ for the full binary basis [Blum, 1984] and $5n - o(n)$ for the binary basis where parity and its complement are excluded [Iwama, Morizumi, 2002]. The properties of the functions under consideration allow us to prove the stated lower bounds with almost no case analysis.

3.8 The Combinatorics of Minimal Unsatisfiability

Oliver Kullmann (Swansea University, GB)

URL http://cs.swan.ac.uk/ csoliver/papers.html#MU2012Dagstuhl

A talk giving an overview on the project of classifying minimally unsatisfiable clause-sets. The basic intuitions behind this project are:

- unsatisfiability of clause-sets can be "explained" by the included minimally unsatisfiable clause-sets
- minimally unsatisfiability can be reduced to basic, intuitive patterns, when using the deficiency (the difference between the number of clauses and the number of variables) as complexity parameter.

The fundamentals are discussed, and then new results, not included in the chapter in the Handbook of Satisfiability, are outlined.

References

1 Armin Biere, Marijn J.H. Heule, Hans van Maaren, and Toby Walsh, editors. *Handbook of Satisfiability*, volume 185 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, February 2009. ISBN 978-1-58603-929-5.

2 Hans Kleine Büning and Oliver Kullmann. Minimal unsatisfiability and autarkies. In [1], chapter 11, pages 339–401. ISBN 978-1-58603-929-5. DOI: 10.3233/978-1-58603-929-5-339.

3.9 Open Problems in Proof Complexity (a Personal Selection)

Massimo Lauria (KTH – Stockholm, SE)

Research in proof complexity focuses on showing lower bounds for stronger and stronger proof systems. Like circuit complexity research, the former has been stuck on difficult open problems for years. The reason is that proof complexity has been invented as a tool for studying computational complexity questions like NP vs coNP.

We propose open problems with a very different motivation. We think that proof complexity approach is useful in an algorithmic setting like combinatorial optimization.

We discuss the problem of finding a k-clique in a graph, using a SAT solver. For $k \ll n$ (as in parameterized complexity theory) we still do not know if there is something better than brute force search to prove that such clique do not exists.

We also discuss the relation between proof systems and approximation. It is known that many approximation algorithms can be proved to be correct in some geometric proof systems: this implies unconditional inapproximability results. While all known lower bounds are rank based, it is open if it is possible to lower bound the length of proofs (i.e. the running time of the algorithms).

3.10 Complexity Classifications for Different Satisfiability Problems

Arne Meier (Leibniz Universität Hannover, DE)

In this talk we introduce the audience to the techniques around Post's lattice [1]. Hereby we define the notion of Boolean clones in terms of a closure operator in means of superposition (introduction of fictive variables, arbitrary composition) applied to a finite set of Boolean functions. We explain how the lattice enables us to state complexity classifications of Boolean problems in a very structured and complete way.

The central motivation of this approach is to understand the inherent structure of a given Boolean problem and the possible connection of the underlying difficulty to a specific set of Boolean functions or, in fact, to the presence of a single Boolean function.

Next we describe several general steps which can be done always when one works with Post's lattice and demonstrate the power of this tool in following Lewis classification of propositional SAT from 1979 in complete detail [2].

Finally we visit the temporal logic CTL (Computation Tree Logic) and give an intuition about how the complexity landscape looks for this logic satisfiability problem which refers to the results in [3].

References

1 E. Post. *The two-valued iterative systems of mathematical logic.* Annals of Mathematical Studies, Vol. 5, pp. 1–122, 1941.

2 H. Lewis. *Satisfiability problems for propositional calculi.* Math. Sys. Theory, Vol. 13, pp. 45–53, 1979.

3 A. Meier and M. Mundhenk and M. Thomas and H. Vollmer. *The Complexity of Satisfiability for Fragments of CTL and CTL^**. International Journal of Foundations of Computer Science, No. 5, Vol. 20, pp. 901–918, 2009.

3.11 A Fragment of Dependence Logic Capturing Polynomial Time

Julian-Steffen Müller (Leibniz Universität Hannover, DE)

Joint work of Ebbing, Johannes; Kontinen, Juha; Müller, Julian-Steffen; Vollmer, Heribert
Main reference J. Ebbing, J. Kontinen, J.-S. Müller, H. Vollmer, "A Fragment of Dependence Logic Capturing Polynomial Time," arXiv:1210.3321v2 [cs.LO].
URL http://arxiv.org/abs/1210.3321

In this talk we study the expressive power of Horn-formulae in dependence logic and show that they can express NP-complete problems. Therefore we define an even smaller fragment D-Horn* and show that over finite successor structures it captures the complexity class P of all sets decidable in polynomial time.

3.12 Relating Proof Complexity Measures and Practical Hardness of SAT

Jakob Nordström (KTH – Stockholm, SE)

Joint work of Järvisalo, Matti; Matsliah, Arie; Nordström, Jakob; Živný, Stanislav
Main reference M. Järvisalo, A. Matsliah, J. Nordström, S. Zivný, "Relating Proof Complexity Measures and Practical Hardness of SAT," in Proc. of the 18th Int'l Conf. on Principles and Practice of Constraint Programming (CP '12), LNCS, Vol. 7514, pp. 316–331, 2012.
URL http://dx.doi.org/10.1007/978-3-642-33558-7_25

Boolean satisfiability (SAT) solvers have improved enormously in performance over the last 10-15 years and are today an indispensable tool for solving a wide range of computational problems. However, our understanding of what makes SAT instances hard or easy in practice is still quite limited. A recent line of research in proof complexity has studied theoretical complexity measures such as length, width, and space in resolution, which is a proof system closely related to state-of-the-art conflict-driven clause learning (CDCL) SAT solvers. Although it seems like a natural question whether these complexity measures could be relevant for understanding the practical hardness of SAT instances, to date there has been very limited research on such possible connections.

This work sets out on a systematic study of the interconnections between theoretical complexity and practical SAT solver performance. Our main focus is on space complexity in resolution, and we report results from extensive experiments aimed at understanding to what extent this measure is correlated with hardness in practice. Our conclusion from the empirical data is that the resolution space complexity of a formula would seem to be a more fine-grained indicator of whether the formula is hard or easy than the length or width needed in a resolution proof. On the theory side, we prove a separation of general and tree-like resolution space, where the latter has been proposed before as a measure of practical hardness, and also show connections between resolution space and backdoor sets.

3.13 Exponential Lower Bounds for the PPSZ k-SAT Algorithm

Dominik Scheder (Aarhus University, DK)

Joint work of Chen Shiteng; Scheder, Dominik; Talebanfard, Navid; Tang, Bangsheng
Main reference S. Chen, D. Scheder, N. Talebanfard, B. Tang, "Expoential Lower Bounds for the PPSZ k-SAT Algorithm," to appear at SODA 2013.
URL http://users-cs.au.dk/dscheder/dscheder-homepage/publications/ChenSchederTalebanfardTang.pdf

In 1998, Paturi, Pudlak, Saks, and Zane presented PPSZ, an elegant randomized algorithm for k-SAT. Fourteen years on, this algorithm is still the fastest known worst-case algorithm. They proved that its expected running time on $k-$CNF formulas with n variables is at most $2^{((1-\epsilon_k)n)}$, where $\epsilon_k = \Omega(1/k)$. So far, no exponential lower bounds at all have been known.

In this paper, we construct hard instances for PPSZ. That is, we construct satisfiable k-CNF formulas over n variables on which the expected running time is at least $2^{((1-\epsilon_k)n)}$, for ϵ_k in $O(log^2(k)/k)$.

3.14 Stochastic Local Search for SAT and Make versus Break

Uwe Schöning (Universität Ulm, DE)

Joint work of Balint, Adrian; Schöning, Uwe
Main reference A. Balint, U. Schöning, "Choosing Probability Distributions for Stochastic Local Search and the Role of Make versus Break," in Proc. of the 15th Int'l Conf. on Theory and Applications of Satisfiability Testing (SAT'12), LNCS, Vol. 7317, pp. 16–29, 2012.
URL http://dx.doi.org/10.1007/978-3-642-31612-8_3

Given an assignment a to a CNF formula and a Boolean variable x, MAKE=MAKE(a, x) is the number of clauses which go from false to true when flipping x's assignment. BREAK= BREAK(a, x) is the number of clauses which go from true to false when flipping x.

It seems natural that MAKE – BREAK is a natural measure to base decisions about selection of flipping variables about it. Our experiments show that BREAK is the more important parameter, actually MAKE can be ignored totally - as long as the flip variable x stems from a clause which is false under the actual assignment a.

Another experimental observation is that a 3SAT algorithm based on flipping probabilities (For those variables x as mentioned above) which are proportional to $(1+\text{BREAK}(a, x))^{-3}$ works very well.

3.15 A Satisfiability-Based Approach for Generalized Tanglegrams on Level Graphs

Ewald Speckenmeyer (Universität Köln, DE)

Joint work of Wotzlaw, Andreas; Speckenmeyer, Ewald; Porschen, Stefan
Main reference A. Wotzlaw, E.Speckenmeyer, S. Porschen, "Generalized k-ary tanglegrams on level graphs: A satisfiability-based approach and its evaluation," Discrete Appl. Math., pp. 2349–2363, Volume 160, Issues 16–17, November 2012.
URL http://dx.doi.org/10.1016/j.dam.2012.05.028

A tanglegram is a pair of (not necessarily binary) trees on the same set of leaves with matching leaves in the two trees joined by an edge. Tanglegrams are widely used in computational biology to compare evolutionary histories of species. In this work we present a formulation of two related combinatorial embedding problems concerning tanglegrams in terms of CNF-formulas. The first problem is known as the planar embedding and the second as the crossing minimization problem. We show that our satisfiability-based encoding of these problems can handle a much more general case with more than two, not necessarily binary or complete, trees defined on arbitrary sets of leaves and allowed to vary their layouts. Furthermore, we present an experimental comparison of our technique and several known heuristics for solving generalized binary tanglegrams, showing its competitive performance and efficiency and thus proving its practical usability.

(Slides: http://e-archive.informatik.uni-koeln.de/id/eprint/693)

3.16 State-based Satisfiability

Sean Weaver (University of Cincinnati, US)

State-based Satisfiability (SAT),a variant of SAT that uses state machines to represent constraints. Using this constraint representation allows for compact representations of SAT problem instances that retain more ungarbled user-domain information than other more common representations such as Conjunctive Normal Form. State-base SAT also supports earlier inference deduction during search, the use of powerful search heuristics, and the integration of special purpose constraints and solvers.

3.17 Finiteness Conjecture on Hitting Unsatisfiable Formulas

Xishun Zhao (Sun Yat-sen University – Guangzhou, CN)

Joint work of Zhao, Xishun; Kullmann, Oliver

In this talk we propose the following so-called finiteness conjecture on Hitting unsatisfiable formulas: For every k, there is a number n_k such that every hitting unsatisfiable formula with deficiency k has at most n_k propositional variables if in the formula every literal occurs at least twice. Here, deficiency of a CNF formula is the difference between the number of clauses and the number of variables. Please note that unsatisfiable hitting formulas must be minimal unsatisfiable (MU), that is, deleting any clause results in a satisfiable formula. From some known results on MU, we can see that the conjecture holds for $k \in \{1, 2\}$. In this talk a proof of the conjecture for $k = 3$ is presented. In the proof the singular DP (for short sDP) reduction plays an important role. sDP reduction is Davis-Putnam resolution applied on a variable which or whose negation occurs only once. Another proof trick is the splitting. If F is a hitting unsatisfiable and x is a variable occurring in F, then $F[x = 1]$ and $F[x = 0]$ are also hitting unsatisfiable.

Participants

- Olaf Beyersdorff
University of Leeds, GB
- Uwe Bubeck
Universität Paderborn, DE
- Catarina Carvalho
University of Hertfordshire, GB
- Nadia Creignou
Université de Marseille, FR
- Stefan Dantchev
University of Durham, GB
- Evgeny Dantsin
Roosevelt Univ. – Chicago, US
- Anuj Dawar
University of Cambridge, GB
- Arnaud Durand
University Paris-Diderot, FR
- Johannes Ebbing
Leibniz Univ. Hannover, DE
- Uwe Egly
TU Wien, AT
- John Franco
University of Cincinnati, US
- Nicola Galesi
University of Rome "La Sapienza", IT
- Heidi Gebauer
ETH Zürich, CH
- Andreas Goerdt
TU Chemnitz, DE
- Miki Hermann
Ecole Polytechnique – Palaiseau, FR
- Timon Hertli
ETH Zürich, CH
- Edward A. Hirsch
Steklov Institute – St. Petersburg, RU
- Kazuo Iwama
Kyoto University, JP
- Jan Johannsen
LMU München, DE
- Lefteris M. Kirousis
University of Athens & Computer Technology Institute & Press "Eudoxus"
- Hans Kleine Büning
Universität Paderborn, DE
- Donald Ervin Knuth
Stanford University, US
- Juha Kontinen
University of Helsinki, FI
- Alexander S. Kulikov
Steklov Institute – St. Petersburg, RU
- Oliver Kullmann
Swansea University, GB
- Massimo Lauria
KTH – Stockholm, SE
- Victor W. Marek
University of Kentucky, US
- Barnaby Martin
Middlesex University, GB
- Arne Meier
Leibniz Univ. Hannover, DE
- Julian-Steffen Müller
Leibniz Univ. Hannover, DE
- Jakob Nordström
KTH – Stockholm, SE
- Ramamohan Paturi
University of California – San Diego, US
- Rahul Santhanam
University of Edinburgh, GB
- Dominik Scheder
Aarhus University, DK
- Henning Schnoor
Universität Kiel, DE
- Uwe Schöning
Universität Ulm, DE
- Martina Seidl
University of Linz, AT
- Robert H. Sloan
Univ. of Illinois – Chicago, US
- Ewald Speckenmeyer
Universität Köln, DE
- Stefan Szeider
Vienna Univ. of Technology, AT
- Jacobo Toran
Universität Ulm, DE
- Heribert Vollmer
Leibniz Univ. Hannover, DE
- Sean Weaver
University of Cincinnati, US
- Xishun Zhao
Sun Yat-sen University – Guangzhou, CN