Report from Dagstuhl Seminar 21361

Extending the Synergies Between SAT and Description Logics

Edited by

Joao Marques-Silva[1], Rafael Peñaloza[2], and Uli Sattler[3]

1 **CNRS – Toulouse, FR,** joao.marques-silva@irit.fr
2 **University of Milano-Bicocca, IT,** rafael.penaloza@unimib.it
3 **University of Manchester, GB,** uli.sattler@manchester.ac.uk

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 21361 "Extending the Synergies Between SAT and Description Logics". Propositional satisfiability (SAT) and description logics (DL) are two successful areas of computational logic where automated reasoning plays a fundamental role. While they share a common core (formalised on logic), the developments in both areas have diverged in their scopes, methods, and applications. The goal of this seminar was to reconnect the SAT and DL communities (understood in a broad sense) so that they can benefit from each other. The seminar thus focused on explaining the foundational principles, main results, and open problems of each area, and discussing potential avenues for collaborative progress.

Seminar September 5–10, 2021 – http://www.dagstuhl.de/21361
2012 ACM Subject Classification Theory of computation → Complexity theory and logic; Theory of computation → Description logics; Theory of computation → Logic and verification
Keywords and phrases description logics, propositional satisfiability, reasoning services, standard and non-standard inferences
Digital Object Identifier 10.4230/DagRep.11.8.1

1 Executive Summary

Rafael Peñaloza (University of Milano-Bicocca, IT)
Joao Marques-Silva (CNRS – Toulouse, FR)
Uli Sattler (University of Manchester, GB)

About the Seminar

Propositional satisfiability (SAT) and Description Logics (DL) are two successful areas of computational logic where automated reasoning plays a fundamental role. Seen from a very abstract level, they can be thought as being part of the same family of logical formalisms attempting to represent knowledge from an application domain, and differentiated only by their expressivity and correspondent trade-off in reasoning complexity. However, the evolution of the two areas has diverged, mainly due to differences in their underlying goals and methods. While the DL community focused on introducing and fully understanding new constructors capable of expressing different facets of knowledge, the SAT community built highly-optimised solvers targeted for industrial-size problems.

Extending the Synergies Between SAT and Description Logics, *Dagstuhl Reports*, Vol. 11, Issue 08, pp. 1–10
Editors: Joao Marques-Silva, Rafael Peñaloza, and Uli Sattler
Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Some recent work has permeated the boundaries between the two communities. It has been shown that some DL reasoning problems could be reduced to known SAT-related tasks. In turn, these reductions motivated new optimisations targeted to the specific shape of the problems constructed by them. The goal of this seminar was to bring together researchers from both communities to foster a deeper collaboration and mutual development. The primary goals were (i) to understand the tasks and methods from one community which could benefit the other, and (ii) to discuss the policies used within the communities to encourage specific advancements.

A relevant issue considered is how to promote the development and testing of DL reasoners. To try to answer this, we discussed the status of benchmarks in both communities, and the success stories from SAT competitions. A salient point was the issue, from the DL point of view, of the many variants that should be evaluated – from the different languages, to the reasoning tasks considered. However, recent SAT competitions have also successfully handled many categories. One possible explanation for the wide availability of solvers capable of handling practical extensions of SAT (like MaxSAT and QBF) is the existence of solvers like MiniSAT, which allow for fast prototyping using SAT solvers as oracles. No analogous tool is available for DL reasoners.

The remaining of the seminar focused on novel and timely tasks which are currently under development in both communities, and where the best possibilities for collaborations are foreseen. Among them, we can mention methods for explaining the result from a solver, and proofs which can be used to automatically verify their correctness. We noted that the notion of an *explanation* is too wide, allowing for different interpretations which were presented as talks during the seminar. Each of these interpretations gives rise to distinct techniques. But interestingly, the core ideas are not necessarily specific to SAT or DLs. This last observation can lead to collaborations studying the problems from both points of view.

In addition to the longer talks whose abstracts accompany this document, other impromptu presentations were triggered by the previous discussions. One clear conclusion which can be taken from these engagements is that the potential for synergic growth between the areas is large and worth exploring.

Format

Due to the COVID-19 situation, the seminar had to be held in a hybrid format. While this had the obvious disadvantage of limiting the social interactions and offline scientific discussions that characterise Dagstuhl seminars, it also allowed the participation of many who, by distance or travel limitations, would have not been able to attend.

Overall, the hybrid format meant having a more structured and linear program than originally planned for the seminar, but as mentioned already the results are promising.

2 Table of Contents

3 Overview of Talks

3.1 SHAP Explanations with Booleans Circuit Classifiers

Leopoldo Bertossi (Adolfo Ibáñez University – Santiago, CL)

The presentation turns around the subject of explainable AI. More specifically, we deal with attribution numerical scores that are assigned to features values of an entity under classification, to identify and rank their importance for the obtained classification label.

We concentrate on the popular SHAP score [2] that can be applied with black-box and open models. We show that, in contrast to its general #P-hardness, it can be computed in polynomial time for classifiers that are based on decomposable and deterministic Boolean decision circuits. This class of classifiers includes decision trees and ordered binary decision diagrams. This result was established in [1]. The presentation illustrates how the proof heavily relies on the connection to SAT-related computational problems.

References

1 Arenas, M., Barceló, P., Bertossi, L. and Monet, M. The Tractability of SHAP-scores over Deterministic and Decomposable Boolean Circuits. Proc. AAAI 2021, pp. 6670-6678.
2 Lundberg , S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J., Nair, B., Katz, R., Himmelfarb, J., Bansal, N. and Lee, S-I. From Local Explanations to Global Understanding with Explainable AI for Trees. Nature Machine Intelligence, 2020, 2(1):56-67.

3.2 SGGS decision procedures for fragments of first-order logic

Maria Paola Bonacina (University of Verona, IT)

Joint work of Maria Paola Bonacina, David A. Plaisted, Sarah Winkler
Main reference Maria Paola Bonacina, Sarah Winkler: "SGGS Decision Procedures", in Proc. of the Automated Reasoning – 10th International Joint Conference, IJCAR 2020, Paris, France, July 1-4, 2020, Proceedings, Part I, Lecture Notes in Computer Science, Vol. 12166, pp. 356–374, Springer, 2020.
URL http://dx.doi.org/10.1007/978-3-030-51074-9_20

SGGS (Semantically-Guided Goal-Sensitive reasoning) is an attractive theorem-proving method for decision procedures, because it generalizes the Conflict-Driven Clause Learning (CDCL) procedure for propositional satisfiability, and it is model-complete in the limit, so that SGGS decision procedures are model-constructing. After summarizing the foundations of SGGS as a theorem-proving method, this talk presents recent and ongoing work on SGGS decision procedures for fragments of first-order logic. This includes both negative and positive results about known decidable fragments: for example, SGGS decides the stratified fragment, and hence Effectively PRopositional logic (EPR). SGGS also allows us to discover several new decidable fragments based on well-founded orderings. For most of these new fragments the small model property holds, as the cardinality of SGGS-generated models can be upper bounded, and membership can be tested by applying termination tools for rewriting. A report on experiments with the prototype theorem prover Koala, which is the first implementation of SGGS, closes the presentation. (SGGS is joint work with David Plaisted; SGGS decision procedures are joint work with Sarah Winkler, who is the author of Koala).

3.3 Clauses and Beyond: On Fast Prototyping with SAT Oracles

Alexey Ignatiev (Monash University – Clayton, AU)

Joint work of Alexey Ignatiev, Antonio Morgado, and Joao Marques-Silva
Main reference Alexey Ignatiev, António Morgado, João Marques-Silva: "PySAT: A Python Toolkit for Prototyping with SAT Oracles", in Proc. of the Theory and Applications of Satisfiability Testing – SAT 2018 – 21st International Conference, SAT 2018, Held as Part of the Federated Logic Conference, FloC 2018, Oxford, UK, July 9-12, 2018, Proceedings, Lecture Notes in Computer Science, Vol. 10929, pp. 428–437, Springer, 2018.
URL http://dx.doi.org/10.1007/978-3-319-94144-8_26

This talk overviews SAT-based modeling capabilities offered by the PySAT toolkit. The toolkit aims at providing a simple and unified interface to a number of state-of-the-art Boolean satisfiability (SAT) solvers as well as to a variety of cardinality and pseudo-Boolean encodings. The purpose of PySAT is to enable researchers working on SAT and its applications and generalizations to easily prototype with SAT oracles in Python while exploiting incrementally the power of the original low-level implementations of modern SAT solvers. The toolkit can be helpful when solving problems in NP but also beyond NP that admit either direct clausal or non-clausal representation.

3.4 Modeling and Solving Problems with SAT

Jean-Marie Lagniez (CNRS, CRIL – Lens, FR)

SAT solvers are nowadays used to tackle a large panel of combinatorial problems. In this talk we highlight three different situations where we show that it is possible to significatively improve the SAT solver effectiveness when considering the problems' nature. First, in [1], we show that by playing on clause database cleaning, assumptions managements and other classical parameters, it was possible to immediately and significantly improve an intensive assumption-based incremental SAT solving task: Minimal Unsatisfiable Set. Second, in [2], we show that it is possible to overcome the difficulty of a problem by encoding it incrementally. We experimentally demonstrate that, by using this trick, the Zykov's encoding can be advantageously leveraged to tackle the graph coloring problem. Finally, we discussed solving the Team Formation problem (TF) with SAT technology. We show that for this problem which consists in solving a set cover problem with a large cardinality constraint, it is more advantageous to leverage a MaxSAT solver rather than a SAT solver. This clearly demonstrates the inefficiency of SAT solvers to deal properly with large cardinality constraints.

References

1 Gilles Audemard, Jean-Marie Lagniez, Laurent Simon: *Improving Glucose for Incremental SAT Solving with Assumptions: Application to MUS Extraction.* SAT 2013: 309-317
2 Gael Glorian, Jean-Marie Lagniez, Valentin Montmirail, Nicolas Szczepanski: *An Incremental SAT-Based Approach to the Graph Colouring Problem.* CP 2019: 213-231
3 Nicolas Schwind, Emir Demirovic, Katsumi Inoue, Jean-Marie Lagniez: *Partial Robustness in Team Formation: Bridging the Gap between Robustness and Resilience.* AAMAS 2021: 1154-1162

3.5 Proofs, Proof Logging, Trust, and Certification

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

Modern combinatorial optimization has had a major impact in science and industry. However, the problems considered are computationally very challenging, requiring increasingly sophisticated algorithm design, and there is a poor scientific understanding of how these complex algorithms, called combinatorial solvers, work. More importantly, even mature commercial solvers are known to sometimes produce wrong results, which can be fatal for some types of applications.

One way to address this problem is to try to enhance combinatorial solvers with proof logging, meaning that they output not only solutions but also proofs of correctness. One can then feed the problem, solution, and proof to a dedicated proof checker to verify that there are no errors. Crucially, such proofs should require low overhead to generate and be easy to check, but should supply 100% guarantees of correctness.

In addition to ensuring correctness, such proof logging could also provide strong development support in that it can quickly flag errors during solver software development. And since the proofs give detailed information about what reasoning steps were performed, this opens up new opportunities for in-depth performance analysis and for identifying potential for further improvements. Finally, it enables auditability by third parties without access to the solver used, and furnishes a stepping stone towards making results explainable.

In this presentation, we review proof logging as it has been adopted by the Boolean satisfiability (SAT) community, and discuss some of the challenges that lie ahead if we want to extend proof logging techniques to more general paradigms in combinatorial optimization. Along the way, we discuss what is meant by a “proof” in a formal sense, and the trade-offs involved between maximizing the efficiency of verification methods and minimizing the need for trust in such methods.

3.6 Provenance in Description Logics

Ana Ozaki (University of Bergen, NO)

Joint work of Camille Bourgaux, Diego Calvanese, Davide Lanti, Ana Ozaki, Rafael Peñaloza, Livia Predoiu, Guohui Xiao
Main reference Diego Calvanese, Davide Lanti, Ana Ozaki, Rafael Peñaloza, Guohui Xiao: “Enriching Ontology-based Data Access with Provenance”, in Proc. of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019, pp. 1616–1623, ijcai.org, 2019.
URL http://dx.doi.org/10.24963/ijcai.2019/224

We address the problem of handling provenance information in description logic ontologies [1, 2, 4, 3, 5]. We consider a setting for ontology-based data access in the classical DL-Lite$_R$ ontology language [3] and a setting for ontology-mediated access for the $\mathcal{ELH}^r$ ontology language [2]. Our works are based on semirings and extend the notion of data provenance in database theory. Here ontology axioms and mappings are also annotated with provenance tokens. A consequence inherits the provenance of the axioms involved in deriving it, yielding a provenance polynomial as annotation. We analyse the semantics for the already mentioned ontology languages DL-Lite$_R$ and $\mathcal{ELH}^r$ and investigate the problems of computing provenance and of determining whether a given expression correctly represents the provenance information of a query. In particular, we show that the presence of conjunctions poses various difficulties for handling provenance, some of which are mitigated by assuming multiplicative idempotency of the semiring. We also analyse the problem of computing the set of relevant axioms for a consequence in the $\mathcal{ELH}^r$ case.

References

1 Camille Bourgaux and Ana Ozaki. Querying attributed DL-Lite ontologies using provenance semirings. In *AAAI*, 2019.

2 Camille Bourgaux, Ana Ozaki, Rafael Peñaloza, and Livia Predoiu. Provenance for the description logic elhr. In Christian Bessiere, editor, *IJCAI*, pages 1862–1869. ijcai.org, 2020.

3 Diego Calvanese, Davide Lanti, Ana Ozaki, Rafael Peñaloza, and Guohui Xiao. Enriching ontology-based data access with provenance. In Sarit Kraus, editor, *IJCAI*, pages 1616–1623. ijcai.org, 2019.

4 Ana Ozaki and Rafael Peñaloza. Provenance in ontology-based data access. In Magdalena Ortiz and Thomas Schneider, editors, *DL*, volume 2211 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2018.

5 Rafael Peñaloza. An upper bound for provenance in elhr. In Martin Homola, Vladislav Ryzhikov, and Renate A. Schmidt, editors, *DL*, volume 2954 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2021.

3.7 ASP, Beyond NP, and Debugging for Explanations?

Francesco Ricca (University of Calabria, IT)

Joint work of Carmine Dodaro, Philip Gasteiger, Kristian Reale, Francesco Ricca, Konstantin Schekotihin, Giovanni Amendola, Mirek Truszczynski
Main reference Giovanni Amendola, Francesco Ricca, Miroslaw Truszczynski: “Beyond NP: Quantifying over Answer Sets”, Theory Pract. Log. Program., Vol. 19(5-6), pp. 705–721, 2019.
URL http://dx.doi.org/10.1017/S1471068419000140

Answer Set Programming (ASP) is a logic programming paradigm featuring a purely declarative language with comparatively high modeling capabilities. ASP can model problems in NP in a compact and elegant way. The availability of efficient implementations, supporting API for programmers makes it suitable for developing applications. ASP implementations are based on SAT technology, and ASP is also a good candidate tool for implementing complex reasoning with Description logics. ASP is a worthy option for modeling several tasks related to explainabilty, which usually require complex modeling capabilities, with comparatively high computational complexity, often beyond NP. However, modeling problems beyond NP with ASP is known to be complicated, on the one hand, and limited to problems in Σ_2^P on the other. Inspired by the way Quantified Boolean Formulas extend SAT formulas to model problems beyond NP, we proposed an extension of ASP that introduces quantifiers over stable models of programs, called ASP(Q) [2]. The definition of ASP(Q) allows for disjunctive programs, thus all the features of the basic language are retained. However, by limiting to normal (or HCF) programs (extended with aggregates and other useful modeling constructs) in ASP(Q), one can take advantage of the classic generate-define-test modular programming methodology and other modeling techniques developed for these best-understood classes of programs to model any problem in the Polynomial Hierarchy. Indeed, the presence of quantifiers allows one to model complex properties in a direct way, and the solutions follow directly from the definition in the natural language of the problem at hand. Despite that ASP features a simple syntax and intuitive semantics, errors are common during the development of ASP programs. For this reason, we proposed a novel debugging approach allowing for interactive localization of bugs in non-ground programs [1]. The debugging approach points the user directly to a set of non-ground rules involved in the bug, which might be refined (up to the point in which the bug is easily identified) by asking the programmer a sequence of questions on an expected answer set. Our debugger exploits techniques that are related to MUS search, and can be a starting point for developing methods for explaining the outcome of reasonings that can be cast in rule-based form.

References

1 Carmine Dodaro, Philip Gasteiger, Kristian Reale, Francesco Ricca, and Konstantin Schekotihin. Debugging non-ground ASP programs: Technique and graphical tools. *Theory Pract. Log. Program.* 19(2):290–316, 2019.

2 Amendola, G.; Ricca, F.; and Truszczynski, M. Beyond NP: quantifying over answer sets. *Theory Pract. Log. Program.* 19(5-6):705–721, 2019

3.8 Parameterised Complexity of SAT and related problems

Stefan Szeider (TU Wien, AT)

It is well understood that not all instances of the propositional satisfiability problem (SAT) have the same computational hardness. The hardness of instances depends on their structural properties. There are mainly two approaches to mathematically capture structure in SAT instances: (A: Correlation) to capture structure that statistically correlates with CDCL SAT solvers' running time, and (B: Causation) to capture structure that provides a rigorous running time guarantee for a SAT algorithm. In this talk, I will discuss the pros and cons of both approaches. I will then survey the main findings on approach B, covering structure based on graphical and syntactic concepts and on hybrid concepts that combine them.

References

1 Marko Samer and Stefan Szeider. Fixed-Parameter Tractability: in Handbook of Satisfiability, Chapter 17, 2nd Edition, 2021, pp. 693–736, ISBN 978-1-64368-160-3.

3.9 Scaling SAT/MaxSAT encodings to large instances with SLIM

Stefan Szeider (TU Wien, AT)

Encoding a combinatorial problem into SAT, to solve it with a SAT solver, is a compelling approach for solving NP-hard problems. However, the encoding often causes a blowup in encoding size, which limits the approach to small instances. The SAT-based Local Improvement Method (SLIM) overcomes this limitation by applying SAT (or MaxSAT) encodings locally to a heuristically computed global solution. In this talk, I will present the general idea of SLIM and illustrate it with two recent applications to Bayesian network structure learning [1] and the induction of small decision trees [2].

References

1 Vaidyanathan Peruvemba Ramaswamy, Stefan Szeider: Turbocharging Treewidth-Bounded Bayesian Network Structure Learning. AAAI 2021: 3895-3903
2 André Schidler, Stefan Szeider: SAT-based Decision Tree Learning for Large Data Sets. AAAI 2021: 3904-3912

3.10 Existing Benchmarks from Description Logics

David Tena Cucala (University of Oxford, GB)

This talk lists several benchmarks for evaluating empirically the performance of reasoners for expressive Description Logics. The first part of the talk discusses curated repositories of ontologies available in the Web, including NCBO BioPortal, AgroPortal, the Oxford Ontology Repository, the Manchester OWL Corpus, and the repository for the OWL Reasoner Evaluation Competition. The second part of the talk describes several synthetic ontology generators, such as LUBM, UOBM, OntoBench, and OWL2Bench. Finally, the talk discusses some limitations of current benchmarks, and expresses some properties desirable in future testing frameworks for DL reasoners.

References

1 Ma, L., Yang, Y., Qiu, Z .and Xie, G., Pan, Y., Liu, S.: Towards a Complete OWL Ontology Benchmark. In: The Semantic Web: Research and Applications. pp. 125–139. Springer Berlin Heidelberg, Berlin, Heidelberg (2006)
2 NCBO BioPortal: `https://bioportal.bioontology.org`
3 AgroPortal: `http://agroportal.lirmm.fr`
4 Oxford Ontology Repository: `http://www.cs.ox.ac.uk/isg/ontologies/`
5 Manchester OWL Corpus: `http://mowlrepo.cs.manchester.ac.uk/datasets/mowlcorp/`
6 Protegé Ontology Library: `https://protegewiki.stanford.edu/wiki/ProtegeOntologyLibrary`
7 Lehigh University Benchmark: `http://swat.cse.lehigh.edu/projects/lubm/`
8 UOBM generator: `https://www.cs.ox.ac.uk/isg/tools/UOBMGenerator/`
9 OntoBench: `http://ontobench.visualdataweb.org`
10 OWL2Bench: `https://github.com/kracr/owl2bench`
11 OWL Reasoner Evaluation Competition: `http://dx.doi.org/10.5281/zenodo.18578`

Participants

- Meghyn Bienvenu
CNRS & University of Bordeaux – Bordeaux, FR
- Maria Paola Bonacina
University of Verona – Verona, IT
- Stefan Borgwardt
TU Dresden – Dresden, DE
- Camille Bourgaux
CNRS & ENS – Paris, FR
- David Carral
INRIA – Sophia Antipolis, FR
- Aaron Eberhart
Kansas State University – Manhattan, US
- Jean-Marie Lagniez
CNRS, CRIL – Lens, FR
- Joao Marques-Silva
CNRS – Toulouse, FR
- Jordi Planes
University of Lleida – Lleida, ES
- Anni-Yasmin Turhan
TU Dresden – Dresden, DE

Remote Participants

- Carlos Ansotegui
University of Lleida – Lleida, ES
- Leopoldo Bertossi
Adolfo Ibáñez University – Santiago, CL
- Olaf Beyersdorff
Universität Jena – Jena, DE
- Emmanuel Hebrard
LAAS – Toulouse, FR
- Ian Horrocks
University of Oxford – Oxford, GB
- Alexey Ignatiev
Monash University – Clayton, AU
- Matti Järvisalo
University of Helsinki – Helsinki, FI
- Oliver Kullmann
Swansea University, GB
- Carlos Mencía
University of Oviedo – Oviedo, ES
- Thomas Meyer
University of Cape Town – Cape Town, ZA
- Jakob Nordström
University of Copenhagen, DK & Lund University, SE
- Magdalena Ortiz
TU Wien – Vienna, AT
- Ana Ozaki
University of Bergen – Bergen, NO
- Rafael Peñaloza
University of Milano-Bicocca – Milano, IT
- Francesco Ricca
University of Calabria – Rende, IT
- Uli Sattler
University of Manchester – Manchester, GB
- Renate Schmidt
University of Manchester – Manchester, GB
- Stefan Szeider
TU Wien – Vienna, AT
- David Tena Cucala
University of Oxford – Oxford, GB
- Cesare Tinelli
University of Iowa – Iowa City, US
- Renata Wassermann
University of Sao Paulo – Sao Paulo, BR