SPECIAL SESIONS

Applications of Computer Algebra - ACA2018

Plenary Sessions

Applications of Computer Algebra – ACA2018
Santiago de Compostela, June 18–22, 2018

Applications of Computer Algebra to Verification and Satisfiability Checking

James H. Davenport[1]

Boolean Satisfiability Checking is one of the paradoxes of computer science: on the one hand it (as 3-SAT) is the quintessential NP-complete hard problem, on the other hand, problems with millions of instances are solved routinely. If we ask for (semi-)algebraic satisfiability over thre als, the quantified worst case complexity becomes doubly exponential. While computer algebraists wrestle with this complexity, the Satisfiability Modulo Theories community has been working away pragmatically, using very different success criteria, and applying their techniques, especially in software and system verification. However, they could learn more from Computer Algebra, and we could learn from them. This talk will outline some of these directions.

[1]Department of Computer Sciences
University of Bath
Bath, United Kingdom
J.H.Davenport@bath.ac.uk

Applications of Computer Algebra – ACA2018
Santiago de Compostela, June 18–22, 2018

SAT Solvers and Computer Algebra Systems: A Powerful Combination for Mathematics

Vijay Ganesh[1]

In recent years we have witnessed a dramatic improvement in the performance of Boolean SAT solvers, despite the fact that the Boolean satisfiability problem is NP-complete [1, 2]. While SAT solvers are powerful combinatorial search algorithms, they are weak when it comes to domain-specific mathematical knowledge. On the other hand, computer algebra systems (CAS) are deep repositories of mathematical knowledge and contain many sophisticated mathematical algorithms. However, computer algebra systems are not as strong at combinatorial search as SAT solvers. Motivated by problems that require both powerful search and deep knowledge, we propose a SAT+CAS combination method that brings together the best of both these worlds aimed at solving problems in combinatorial mathematics.

In this talk I will present a SAT+CAS system, MathCheck [3, 4], that we developed and used to counterexample many combinatorial conjectures, most notably the Williamson conjecture. I will discuss the internals of MathCheck, how it can be used, and most importantly, how mathematicians can extend such SAT+CAS tools to tackle a variety of problems. I will also argue that we are witnessing a new long-term paradigmatic shift, wherein, previously unrelated methods such as solvers and CAS are being profitably combined to tackle hard mathematical problems.

Keywords: Boolean SAT solvers, Computer algebra systems, Combinatorial mathematics

Mathematics Subject Classification 2010: 68, 05

References

[1] Jia Hui Liang, Vijay Ganesh, Pascal Poupart, and Krzysztof Czarnecki. Learning Rate Based Branching Heuristic for SAT Solvers. In Proceedings of the 19^{th} International Conference on the Theory and Applications of Satisfiability Testing (SAT 2016), Bordeaux, France, July 5-8, 2016.
`https://sites.google.com/a/gsd.uwaterloo.ca/maplesat/`

[2] Jia Hui Liang, Hari Govind V K, Pascal Poupart, Krzysztof Czarnecki, and Vijay Ganesh. An Empirical Study of Branching Heuristics Through the Lens of Global Learning Rate. In Proceedings of the 20^{th} International Conference on Theory

and Applications of Satisfiability Testing (SAT 2017), Melbourne, Australia, Aug 28 - Sep 1, 2017.
https://sites.google.com/a/gsd.uwaterloo.ca/maplesat/

[3] Curtis Bright, Ilias Kotsireas, and Vijay Ganesh. A SAT+CAS Method for Enumerating Williamson Matrices of Even Order. In the Proceedings of the 32^{nd} AAAI Conference on Artificial Intelligence (AAAI 2018), New Orleans, USA, Feb 2-7, 2018.
https://sites.google.com/site/uwmathcheck/

[4] Ed Zulkoski, Curtis Bright, Albert Heinle, Ilias Kotsireas, Krzysztof Czarnecki, and Vijay Ganesh. Combining SAT Solvers with Computer Algebra Systems to Verify Combinatorial Conjectures. Journal of Automated Reasoning (JAR 2017), Volume 58, number 3, pages 313-339, 2017.
https://sites.google.com/site/uwmathcheck/

[1]Electrical and Computer Engineering Department
University of Waterloo
200 University Avenue West, Waterloo, Ontario, Canada N2L 3G1
Email: vganesh@uwaterloo.ca
Website: https://ece.uwaterloo.ca/~vganesh

Applications of Computer Algebra – ACA2018
Santiago de Compostela, June 18–22, 2018

Dealing with real algebraic curves and surfaces for discovery: from experiments to theory and applications

Laureano Gonzalez-Vega[1]

Geometric entities such as the set of the real zeros of a bivariate equation or the image in of a rational parametrization can be treated algorithmically in a very efficient way by using a mixture of symbolic and numerical techniques. This implies that it is possible to know exactly which is the topology (connected components and their relative position, connectedness, singularities, etc.) of such a curve or surface if their equations are known in an exact manner (whatever this means) for moderate or high degrees. We would describe several different "experiments" coming from Algebraic Geodesy and Computer Aided Design that highlight how new visualisation tools in Computational Mathematics mixing symbolic and numerical techniques allow to perform experiments conveying either to mathematical discoveries and/or to new computational techniques useful in applications.

[1]Departamento de Matemáticas, Estadística y Computación
Universidad de Cantabria, Spain

laureano.gonzalez@unican.es

Applications of Computer Algebra – ACA2018
Santiago de Compostela, June 18–22, 2018

Automatic Geometric Theorem Proving and Discovering Using (Comprehensive) Groebner Bases

Dingkang Wang[1]

Automatic geometric theorem proving and discovering is to prove and derive mathematical theorems by computer programs, which has been studied for several decades. It can be traced back to the great work of Tarski, Seidenberg, Gelernter,Collins,Wu and so on. The extensive study in this research field is due to the introduction of Wu's method in later 1970s, which is surprisingly efficient for proving difficult geometric theorem. First, I will introduce our work on discovering geometric theorems by using the comprehensive Groebner systems, i.e. finding some complementary conditions such that the geometric statement will become true under the original hypotheses and these complementary conditions. Particularly, efficient algorithms for computing comprehensive Groebner systems/bases are also reviewed. Second, I will investigate the problem whether the conclusion is true on some components of the hypotheses for a geometric statement. In that case, the affine variety associated with the hypotheses is reducible. A polynomial vanishes on some but not all the components of a variety if and only if it is a zero divisor in a quotient ring with respect to the radical ideal defined by the variety. Based on this fact, we present an algorithm to decide if a geometric statement is only true on components. Besides proving theorems, the parametrical extension of this method can also be used to discover new geometric theorems. That is, we can find out complementary conditions such that the geometric statement becomes true or true on components. Some illustrative examples will be presented to show how the method works.

This is joint work with Deepak Kapur, Yao Sun and Jie Zhou.

Keywords: Automatic Proving and Discovering, Geometric Theorem, Groebner Bases

[1]KLMM, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China
School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing 100049, China

dwang@mmrc.iss.ac.cn