My Personal SAT Solving History

1960
1970
1980
1990
2000
2010
2020

- SAT
- NP complete
- DPLL
- SAT for Planning
- Look Ahead
- 1st SAT Competition
- WalkSAT
- GSAT
- Tseitin Encoding
- CDCL
- Bounded Variable Elimination
- Portfolio
- Phase Saving
- ProBSAT
- Avatar
- Arithmetic Solvers
- Massively Parallel
- QBF working
- SAT & SMT everywhere

- Handbook of SAT (1st)
- Handbook of SAT (2nd)
- SAT Chapter Donald Knuth
- Inprocessing Cube & Conquer
- Cube & Conquer
- Proofs
- SMT
- LBD
- VSIDS
- MiniSAT
- ProbSAT
- Avatar
- Arithmetic Solvers


SAT & SMT everywhere
Massively Parallel
QBF working
Satisfiability (SAT) related topics have attracted researchers from various disciplines. Logic, applied areas such as planning, scheduling, operations research, and combinatorial optimization, but also theoretical issues on the theme of complexity, and much more, they all are connected through SAT.

My personal interest in SAT stems from actual solving. The increase in power of modern SAT solvers over the past 15 years has been phenomenal. It has become the key enabling technology in automated verification of both computer hardware and software. Bounded Model Checking (BMC) of computer hardware is now probably the most widely used model checking technique. The counterexamples it finds are just satisfying instances of a Boolean formula obtained by unwinding to some fixed depth a sequential circuit and its specification in Linear Temporal Logic. Extending model checking to software verification is a much more difficult problem on the frontier of current research. One promising approach for languages like C with finite word-length integers is to use the same idea as in BMC but with a decision procedure for the theory of bit-vectors instead of SAT. All decision procedures for bit-vectors that I am familiar with ultimately make use of a fast SAT solver to handle complex formulas.

Decision procedures for more complicated theories, like linear real and integer arithmetic, are also used in program verification. Most of them use powerful SAT solvers in an essential way.

Clearly, efficient SAT solving is a key technology for 21st century computer science. I expect this collection of papers on all theoretical and practical aspects of SAT solving will be extremely useful to both students and researchers and will lead to many further advances in the field.

Edmund Clarke

Edmund M. Clarke, FOR Systems University Professor of Computer Science and Professor of Electrical and Computer Engineering at Carnegie Mellon University, is one of the initiators and main contributors to the field of Model Checking, for which he also received the 2007 ACM Turing Award.

In the late 90s, Professor Clarke was one of the first researchers to realize that SAT solving has the potential to become one of the most important technologies in model checking.
### Part I. Theory and Algorithms

- John Franco, John Martin: *A History of Satisfiability*. 3-74
- Steven David Prestwich: *CNF Encodings*. 75-97
- Adnan Darwiche, Knot Pipatsrisawat: *Complete Algorithms*. 99-130
- Marijn Heule, Hans van Maaren: *Look-Ahead Based SAT Solvers*. 155-184
- Oliver Kullmann: *Fundamentals of Branching Heuristics*. 205-244
- Dimitris Achlioptas: *Random Satisfiability*. 245-270
- Carla P. Gomes, Ashish Sabharwal: *Exploiting Runtime Variation in Complete Solvers*. 271-288
- Kareem A. Sakallah: *Symmetry and Satisfiability*. 289-338
- Hans Kleine Büning, Oliver Kullmann: *Minimal Unsatisfiability and Autarkies*. 339-401
- Marko Samer, Stefan Szeider: *Fixed-Parameter Tractability*. 425-454

### Part II. Applications and Extensions

- Armin Biere: *Bounded Model Checking*. 457-481
- Jussi Rintanen: *Planning and SAT*. 483-504
- Daniel Kroening: *Software Verification*. 505-532
- Hantao Zhang: *Combinatorial Designs by SAT Solvers*. 533-568
- Fabrizio Amtarelli, Rémi Monasson, Guilhem Semerjian, Francesco Zamponi: *Connections to Statistical Physics*. 569-611
- Chu Min Li, Felip Manyá: *MaxSAT, Hard and Soft Constraints*. 613-631
- Rolf Drechsler, Tommi A. Junttila, Ilkka Niemelä: *Non-Clausal SAT and ATPG*. 655-693
- Olivier Roussel, Vasco M. Marquinho: *Pseudo-Boolean and Cardinality Constraints*. 695-733
- Hans Kleine Büning, Uwe Bubeck: *Theory of Quantified Boolean Formulas*. 735-760
- Enrico Giunchiglia, Paolo Marin, Massimo Narizzano: *Reasoning with Quantified Boolean Formulas*. 761-780
- Roberto Sebastiani, Armando Tacchella: *SAT Techniques for Modal and Description Logics*. 781-824
- Clark W. Barrett, Roberto Sebastiani, Sanjit A. Seshia, Cesare Tinelli: *Satisfiability modulo Theories*. 825-885
- Stephen M. Majercik: *Stochastic Boolean Satisfiability*. 887-925
NEWLY AVAILABLE SECTION OF
THE CLASSIC WORK

The Art of
Computer Programming

VOLUME 4
Satisfiability

FASCICLE 6
Special thanks are due to Armin Biere, Randy Bryant, Sam Buss, Niklas Eén, Ian Gent, Marijn Heule, Holger Hoos, Svante Janson, Peter Jeavons, Daniel Kroening, Oliver Kullmann, Massimo Lauria, Wes Pegden, Will Shortz, Carsten Sinz, Niklas Sörensson, Udo Wermuth, Ryan Williams, and... for their detailed comments on my early attempts at exposition, as well as to numerous other correspondents who have contributed crucial corrections. Thanks also to Stanford’s Information Systems Laboratory for providing extra computer power when my laptop machine was inadequate.

* * *

Wow—Section 7.2.2.2 has turned out to be the longest section, by far, in *The Art of Computer Programming*. The SAT problem is evidently a “killer app,” because it is key to the solution of so many other problems. Consequently I can only hope that my lengthy treatment does not also kill off my faithful readers! As I wrote this material, one topic always seemed to flow naturally into another, so there was no neat way to break this section up into separate subsections. (And anyway the format of *TAOCP* doesn’t allow for a Section 7.2.2.2.1.)

I’ve tried to ameliorate the reader’s navigation problem by adding subheadings at the top of each right-hand page. Furthermore, as in other sections, the exercises appear in an order that roughly parallels the order in which corresponding topics are taken up in the text. Numerous cross-references are provided
editors Armin Biere, Marijn Heule, Hans van Maaren, Toby Walsh

with many updated chapters and the following 7 new chapters:

Proof Complexity Jakob Nordström and Sam Buss

Preprocessing Armin Biere, Matti Järvisalo and Benjamin Kiesl

Tuning and Configuration Holger Hoos, Frank Hutter and Kevin Leyton-Brown

Proofs of Unsatisfiability Marijn Heule

Core-Based MaxSAT Fahiem Bacchus, Matti Järvisalo and Ruben Martins

Proof Systems for Quantified Boolean Formulas Olaf Beyersdorff, Mikoláš Janota, Florian Lonsing and Martina Seidl

Approximate Model Counting Supratik Chakraborty, Kuldeep S. Meel, and Moshe Y. Vardi

SAT solving is a key technology for 21st century computer science.

Edmund Clarke
2007 ACM Turing Award Recipient

The SAT problem is evidently a killer app, because it is key to the solution of so many other problems.

Donald Knuth
1974 ACM Turing Award Recipient

The SAT problem is at the core of arguably the most fundamental question in computer science: What makes a problem hard?

Stephen Cook
1982 ACM Turing Award Recipient
Handbook of Satisfiability
Second Edition

Editors: A. Biere, M. Heule, H. van Maaren, T. Walsh
Volume 336 of Frontiers in Artificial Intelligence and Applications

Propositional logic has been recognized throughout the centuries as one of the cornerstones of reasoning in philosophy and mathematics. Over time, its formalization into Boolean algebra was accompanied by the recognition that a wide range of combinatorial problems can be expressed as propositional satisfiability (SAT) problems. Because of this dual role, SAT developed into a mature, multi-faceted scientific discipline, and from the earliest days of computing a search was underway to discover how to solve SAT problems in an automated fashion.

This book, the Handbook of Satisfiability, is the second, updated and revised edition of the book first published in 2009 under the same name. The handbook aims to capture the full breadth and depth of SAT and to bring together significant progress and advances in automated solving. Topics covered span practical and theoretical research on SAT and its applications and include search algorithms, heuristics, analysis of algorithms, hard instances, randomized formulae, applications, solvers, simplifiers, tools, case studies and empirical results. SAT is interpreted in a broad sense, so as well as propositional satisfiability, there are chapters covering the domain of quantified Boolean formulae (QBF), constraints programming techniques (CSP) for word-level problems and their propositional encoding, and satisfiability modulo theories (SMT). An extensive bibliography completes each chapter.

This second edition of the handbook will be of interest to researchers, graduate students, final-year undergraduates, and practitioners using or contributing to SAT, and will provide both an inspiration and a rich resource for their work.

Edmund Clarke, 2007 ACM Turing Award Recipient: “SAT solving is a key technology for 21st century computer science.”

Donald Knuth, 1974 ACM Turing Award Recipient: “SAT is evidently a killer app, because it is key to the solution of so many other problems.”

Stephen Cook, 1982 ACM Turing Award Recipient: “The SAT problem is at the core of arguably the most fundamental question in computer science: What makes a problem hard?”

For more information and ordering check tiny.cc/SAT2021

Discount Code
Order your print book before April 15, 2021 and get 35% off!
Code: SAT2021

ISBN: 978-1-64368-161-0
€200 / US$250 / £180 excl. VAT

For ebooks, go to www.ebooks.iospress.com

Visit our website www.iospress.com for online ordering
For ebooks, go to www.ebooks.iospress.com
Follow us on Twitter: @IOSPress_STM
Follow us on Facebook: publisheriospress

IOS Press
Nieuwe Herengracht 68
1013 BG Amsterdam
The Netherlands
Tel. +31 20 688 3355
Email: order@iospress.nl

IOS Press, Inc.
6751 Tepper Drive
Clifton, VA 20124
USA
Tel. +1 703 830 6300
Email: sales@iospress.com

Contents
Part I. Theory and Algorithms
Chapter 1. A History of Satisfiability
John Franco and John Martin, with sections contributed by Miguel Arias, Holger Hoos, Hans Kleine Büning, Ewald Speckenmeyer, Alasdair Urquhart, and Hantao Zhang

Chapter 2. CNF Encodings
Steven Prestwich

Chapter 3. Complete Algorithms
Adnan Darwiche and Knot Papitsisawat

Chapter 4. Conflict-Driven Clause Learning SAT Solvers
Joao Marques-Silva, Ines Lynce, and Sharad Malik

Chapter 5. Look-Ahead Based SAT Solvers
Marijn J.H. Heule and Hans van Maaren

Chapter 6. Incomplete Algorithms
Henry Kautz, Ashish Sabharwal, and Bart Selman

Chapter 7. Proof Complexity and SAT Solving
Sam Buss and Jakob Nordström

Chapter 8. Fundaments of Branching Heuristics
Oliver Kullmann

Chapter 9. Preprocessing in SAT Solving
Carla P. Gomes and Ashish Sabharwal

Chapter 10. Random Satisfiability
Holger H. Hoos, Frank Hutter, and Kevin Leyton-Brown

Chapter 11. Exploiting Runtime Variation in Complete Solvers
Carla P. Gomes and Ashish Sabharwal

Chapter 12. Automated Configuration and Selection of SAT Solvers
Holger H. Hoos, Frank Hutter, and Kevin Leyton-Brown

Chapter 13. Symmetry and Satisfiability
Karem A. Sakallah

Chapter 14. Minimal Unsatisfiability and Autarkies
Hans Kleine Büning and Oliver Kullmann

Chapter 15. Proofs of Unsatisfiability
Marijn J.H. Heule

Chapter 16. Worst-Case Upper Bounds
Evgeny Dantsin and Edward A. Hirsch

Chapter 17. Fixed-Parameter Tractability
Markus Samer(1977-2010) and Stefan Szeider

Part II. Applications and Extensions
Chapter 18. Bounded Model Checking
Armin Biere

Chapter 19. Planning and SAT
Jussi Rintanen

Chapter 20. Software Verification
Daniel Kroening

Chapter 21. Combinatorial Designs by SAT Solvers
Hantao Zhang

Chapter 22. Connections to Statistical Physics
Fabrizio Areatelli, Rémi Monasson, Guilhem Semerjian and Francesco Zamponi

Chapter 23. MaxSAT, Hard and Soft Constraints
Chu Min Li and Filip Mariy

Chapter 24. Maximum Satisfiability
Fahiem Bacchus, Matti Järvisalo, and Ruben Martins

Chapter 25. Model Counting
Carla P. Gomes, Ashish Sabharwal, and Bart Selman

Chapter 26. Approximate Model Counting
Supratik Chakrabarty, Kuileep S. Meel, and Moshe Y. Vardi

Chapter 27. Non-Clausal SAT and ATPG
Rolf Drechsler, Tommi Junttila and Ilkka Niemelä

Chapter 28. Pseudo-Boolean and Cardinality Constraints
Olivier Roussel and Vasco Manquinho

Chapter 29. Theory of Quantified Boolean Formulas
Hans Kleine Büning and Uwe Biere

Chapter 30. Reasoning with Quantified Boolean Formulas
Enrico Giunchiglia, Paolo Marin, and Massimo Narizzano

Chapter 31. Quantified Boolean Formulas
Olf Beyersdorff, Mikoláš Janota, Florian Lonsing, Martina Seidl

Chapter 32. SAT Techniques for Modal and Description Logics
Roberto Sebastiani and Armando Tacchella

Chapter 33. Satisfiability Modulo Theories
Supratik Chakraborty, Kuldeep S. Meel, and Moshe Y. Vardi

Chapter 34. Stochastic Boolean Satisfiability
Stephen M. Majercik

Subject Index
Cited Authors Index
Contributing Authors and Affiliations
SAT-Race 2010 Award
“Plingeling” by Armin Biere
is awarded the title of
Best Parallel Solver

SAT-Race 2010 Award
“Lingeling” by Armin Biere
is awarded the title of
Second Prize Winner

Competition 2007
Gold medal

Competition 2009
3rd medal

Eighth International Conference on Theory and Applications of Satisfiability Testing
SAT-Race 2015
Parallel Track
3rd Prize

Eighth International Conference on Theory and Applications of Satisfiability Testing
SAT-Race 2015
Main Track
Special Prize: Most Innovative Solver

Competition 2009
Silver medal

Competition 2009
Gold medal

Competition 2002
Winner
SAT Competition Winners on the SC2020 Benchmark Suite

data produced by Armin Biere and Marijn Heule
some recent Tweets

SAT solvers get faster and faster: all-time winners of the SAT Competition on 2020 instances, featuring our new solver Kissat (fmv.jku.at/kissat), which won in 2020. The web page also has runtime CDFs for 2011 and 2019.

The largest ones have millions of variables and clauses. The planning track had even larger ones. See the variable and clause distribution plot for the main track:

Eventually I will need to support 64-bit variable indices (Lingeling has 2^27, CaDiCaL indeed 2^31 and Kissat 2^28 as compromise though it could easily do half a billion)
This week, we will have again many great talks about SAT at @SimonsInstitute: I am especially looking forward the one of @ArminBiere titled « A Personal History of Practical SAT Solving » simons.berkeley.edu/talks/tbd-308

Did you know that @ArminBiere participated to every SAT competitive event since 2002? And won in a fair amount of tracks? See all his solvers here: fmv.jku.at/software/index...

I remember the solvers Limmat, Compsat, NanoSAT, PicoSAT, PrecoSAT, Lingeling, Splatzt, CaDiCaL, Kissat plus « educational » solvers Cleaneling and Satch. Would love to see a cactus plot with all those solvers on Thursday @ArminBiere :)
My Personal SAT Solving History

- SAT NP complete
- DPLL
- DP
- Look Ahead
- WalkSAT
- GSAT
- Tseitin Encoding
- CDCL
- Bounded Model Checking
- VSIDS
- SMT
- Inprocessing
- Cube & Conquer
- SAT Chapter
- Donald Knuth
- 1st SAT Competition
- 1970
- 1980
- 1990
- 2000
- Portfolio
- Bounded Variable Elimination
- LBD
- MiniSAT
- ProBSAT
- Avatar
- Arithmetic Solvers
- Massively Parallel
- SAT & SMT everywhere
- QBF working
- Handbook of SAT (1st)
- Handbook of SAT (2nd)
- 2020
- 2010
- Phase Saving
- ProBSAT
- 1960
A Personal History of Practical SAT Solving

*Thursday, May 13th, 2021 8:30 am – 9:30 am*

Event: 50 Years of Satisfiability: The Centrality of SAT in the Theory of Computing

**Speaker:** Armin Biere (Johannes Kepler University)

My tutorial and invited talks on SAT solving usually end with a slide showing a timeline of what I consider important milestones in the history of practical SAT solving. As a consequence there is usually not much time left to discuss this history. On this occasion, an invited talk at the workshop on 50 years of satisfiability, there seems to be no better fit for a topic, to turn this around and start by going through those perceived important contributions, from early work in the fifties, over the SAT revolution we are witnessing today, to future challenges. The talk will be structured slightly differently than usual and will be open to frequent interaction with the audience.
Talks Armin Biere

List of some recent public talks.

Talks 2021

A Personal History of Practical SAT Solving
Workshop 50 Years of Satisfiability: The Centrality of SAT in the Theory of Computing
Program Satisfiability: Theory, Practice, and Beyond
Simons Institute for the Theory of Computing
Cyberspace, May 13, 2021
[ video ]

Tutorial
SAT-Solving
Satisfiability: Theory, Practice, and Beyond Boot Camp
Program Satisfiability: Theory, Practice, and Beyond
Simons Institute for the Theory of Computing
Cyberspace, February 2, 2021
[ pre-recorded intro on YouTube (4h22) | local copy of pre-recorded video (4h22) | live recording on YouTube (2h02) | code ]
My Personal SAT Solving History

1960: DPLL
1970: SAT NP complete
1980: Look Ahead, WalkSAT, GSAT
1990: Bounded Variable Elimination, SAT Competition
2000: Portfolio, Bounded Variable Elimination, CDCL
2010: SAT & SMT everywhere, Massively Parallel
2020: QBF working

Handbook of SAT (1st)
SAT Chapter Donald Knuth

Dsolv, Minisat, Probsat, Avat, Cube & Conquer