Handbook of Model Checking by Edmund M. Clarke, Thomas A. Henzinger, Helmut Veith, and Roderick Bloem (eds), published by Springer International Publishing AG, Cham, Switzerland, 2018.

Igor Konnov

Under consideration for publication in Formal Aspects of Computing

Handbook of Model Checking

By Edmund M. Clarke, Thomas A. Henzinger, Helmut Veith, and Roderick Bloem

Springer International Publishing AG, Cham, Switzerland, 2018, xxiv+1210 pp, ISBN 978-3-319-10574-1 (Hardcover, 2.13 kg), ISBN 978-3-319-10575-8 (eBook, PDF).

Reviewed by Igor Konnov

INRIA Nancy — Grand Est
54000 Nancy, France

This book is a comprehensive guide on model checking. Pioneered by Edmund M. Clarke, E. Allen Emerson, and Joseph Sifakis in the early 1980es, this technology brought us tools for automatic analysis of hardware and software, sequential and concurrent. The discovery of model checking was distinguished with the Turing Award in 2007. Although by that time, the technology made a giant leap towards automation of software verification, the classical textbook on model checking [CGP01] covered only the early generation techniques such as explicit-state enumeration and binary decision diagrams. (We were lucky to get the Second Edition of this book [CJGK$^+$18] in 2018 too!) For long time, to learn about these new techniques one had to board a plane and fly to the conference on Computer-Aided Verification, in order to get fresh pointers to the literature. The "Handbook of Model Checking" fills this gap.

This book is invaluable to the PhD students, who like to dive in the field, learn about state-of-the-art techniques and find open problems. It is also helpful to the researchers who like to refresh their memory on a particular model checking technique and get pointers to the further reading. The Handbook contains chapters of varying reading difficulty. While some chapters combine intuition with rigor, other chapters may require careful reading and attention to the details. In any case, as model checking borrows from multiple fields of mathematics and theoretical computer science — e.g., logic, discrete mathematics, graph theory, game theory — the reader should be ready to grasp new concepts and ideas.

Under one cover, the Handbook collects virtually all the research directions in which model checking went since its invention. Chapters 1–8 start with the classics: Kripke structures, temporal logics, Büchi automata, explicit-state model checking, partial-order reduction, and symbolic model checking with binary decision diagrams. Although this technical material can be also found in [CGP01, BK08], in the Handbook, the authors often put the techniques in the historical perspective and share their experience on what works well and what does not. These chapters are complimented by Chapter 23 — a hidden gem by Robert Kurshan on "Transfer of Model Checking to Industrial Practice". To those who seek motivation for studying this topic, I would recommend reading Chapter 23 right after Chapter 1.

Chapters 9 and 11 introduce the SAT and SMT solvers, which are nowadays used as reasoning back-ends of many model checkers. An in-depth discussion of SAT and SMT solvers can be found in [BHvMW09] and [KS16]. Chapters 10, 12, 13–15 introduce the mainstream techniques that apply SAT and SMT for efficient model checking of software and hardware: bounded model checking, compositional reasoning, abstraction and abstraction refinement, predicate abstraction, and interpolation. Further, Chapters 17–18 discuss

Correspondence and offprint requests to: I. Konnov

the approaches for dealing with procedures and concurrency in software model checking. These techniques mainly built upon the framework of push-down automata.

Chapters 16, 19, 20 build the bridges between model checking and other approaches to ensuring system correctness: data-flow analysis and abstract interpretation, automated theorem proving, and automated testing. Although these approaches were initially thought of as orthogonal, they are much closer now, due to cross fertilization.

Chapters 21, 22, 28, 30, 31 introduce the techniques that bring model checking beyond the standard application domains. Whereas the classical model checking deals with finite-state Kripke structures, parameterized model checking answers the question, whether a temporal property holds for the systems comprising an *arbitrary number* of components. Chapter 21 presents several prominent techniques that answer this question for special kinds of parameterized protocols. A recent survey on this topic has also appeared in [BJK+15]. Chapter 22 introduces model checking techniques for *security protocols*. These protocols are modeled as term transformers in the Dolev-Yao model. Chapter 28 discusses model checking of probabilistic systems. In these systems, events happen with certain probabilities. Instead of checking, whether such a system satisfies a temporal property, one is interested in answering the question of whether a temporal property holds with a given *probability*. Chapter 30 presents techniques for hybrid systems, in which some variables may change continuously, and their evolution is constrained with differential equations. Chapter 31 deals with model checking of *infinite-state* systems by fix-point computations over regions in the framework of μ-calculus.

Chapters 23–25 introduce applications of model checking in hardware industry. Chapter 23 summarizes the success of model checking in hardware verification and pitfalls that one meets when transferring theoretical results in practice. Chapter 24 discusses practical specification languages that are used in hardware industry for expressing the expected behavior of a hardware circuit. Related to that, Chapter 25 presents the technique of symbolic trajectory evaluation, which is used in industry for circuit verification.

Chapters 26, 29, 32 discuss the well-studied extensions of model checking techniques μ-calculus, timed automata, and process algebras. Finally, Chapter 27 presents the techniques for synthesis of reactive systems, as opposite to verification.

Obviously, it is hard to imagine that one reads the Handbook from cover to cover. When this book was lying on my desk for several months, I would browse through chapters on this or that topic. What I like the most about the book is that many chapters contain contributions by authors from different teams. This gives the reader a balanced view on every approach.

As the large part of the coordination and copy-editing was shared among three teams in Austria, including Helmut Veith's group, I was lucky to witness the work on the Handbook. This was a long-term project that could be only finished thanks to the energy of the editors and contributions by the authors, reviewers, and the editing team. To a shock to all of us, Helmut Veith tragically passed away, when the Handbook was in the final stage. We are missing his brilliance, energy, and unordinary thinking.

Finally, I have to give a few comments about the format of the Handbook. Owing to the high-quality paper and its 1234 pages, the hardcover is truly heavy, it weighs over two kilograms. Although it looks amazing on a bookshelf, it is rather an exercise to hold the Handbook in hands for longer than five minutes. I would thus recommend buying the eBook and printing the chapters of interest, when needed.

References

[BHvMW09] Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors. *Handbook of Satisfiability*, volume 185 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2009.

[BJK+15] Roderick Bloem, Swen Jacobs, Ayrat Khalimov, Igor Konnov, Sasha Rubin, Helmut Veith, and Josef Widder. *Decidability of Parameterized Verification*. Synthesis Lectures on Distributed Computing Theory. Morgan & Claypool Publishers, 2015.

[BK08] Christel Baier and Joost-Pieter Katoen. *Principles of model checking*. MIT Press, 2008.

[CGP01] Edmund M. Clarke, Orna Grumberg, and Doron A. Peled. *Model checking*. MIT Press, 2001.

[CJGK+18] Edmund M. Clarke Jr, Orna Grumberg, Daniel Kroening, Doron Peled, and Helmut Veith. *Model checking*. MIT press, 2018. Second Edition.

[KS16] Daniel Kroening and Ofer Strichman. *Decision Procedures - An Algorithmic Point of View, Second Edition*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2016.