Book Review:*

Verification of Reactive Systems: Formal Methods and Algorithms

by Klaus Schneider

(Springer-Verlag, 2004)

Joël Ouaknine
Oxford University Computing Laboratory
joel@comlab.ox.ac.uk

This book targets three types of audience: theoreticians, theoreticians, and theoreticians. As a member of the latter, I welcome it with open arms, but practitioners—and, I suspect, many readers of *Software Testing, Verification and Reliability*—beware: Klaus Schneider's *Verification of Reactive Systems* very much deals with the fundamental theory of its subject matter, namely automata, temporal logic, and model checking.

In that respect, however, it is an admirable piece of work. It begins with a historical overview of formal methods (going as far back as Descartes, Pascal, and Leibniz!), and then proposes a uniform and wide-ranging notation that is used throughout the book. Chapter 3 treats fixpoint calculi, and especially the (vectorized) μ-Calculus. It goes in great detail into the model checking algorithms, including their associated complexity. Chapter 4 is devoted to ω-automata, covering (with full proofs) many of the standard results on the subject, including Safra's determinization construction, the Borel hierarchy, and model checking procedures. The next chapter treats temporal logics, and in particular extensions of CTL. Finally, Chapter 6 studies first- and second-order predicate logic, mainly insofar as it relates to the previous items.

The book, therefore, is essentially centered around four themes: the μ-Calculus, ω-automata, temporal logic, and predicate logic. It answers just about any question one might ask concerning the various interrelationships among these topics, with a noted algorithmic and complexity-theoretic slant. Moreover, not only are proofs of most results given in the text, but extensive references to the literature—the bibliography has over five hundred entries—make it an extremely valuable research tool.

Three appendices conclude the presentation, on binary decision diagrams,

*Appeared in *Software Testing, Verification and Reliability* 15, 2005.

local model checking and satisfiability checking algorithms for the μ-Calculus, and abstraction techniques. The underlying aim of these appendices, apart from completeness, is to introduce some of the data structures and optimizations one might wish to make use of to efficiently implement the algorithms described in the core. The treatment remains resolutely theoretical, and the serious implementer would probably be best advised to follow the numerous pointers to the literature rather than rely exclusively on the text.

In summary, *Verification of Reactive Systems* is both focused (for example, alternating and tree automata are completely left out), systematic, and thorough. This makes it very suitable for a graduate course in selected topics in formal methods. It would also be highly useful as a general reference in logic and automata theory, provided readers are willing to invest the time to familiarize themselves with Schneider's uniform technical terminology: indeed, the book's notation often differs from that of the primary sources, and unless one is conversant with the former, reconciling the text with the literature is likely to prove quite arduous.

A minor, if irritating, defect of the book is its poor editing. The text abounds with typographical, grammatical, and stylistic errors, many of which could easily have been avoided (what spell-checker would fail to catch such a conspicuous neologism as "*suggerates*"?). Repetitions and redundancies are also not infrequent (although given the sheer amount of material covered, one might view this as a feature rather than a bug!). This being said, these relatively small flaws rarely impede understanding. Overall, *Verification of Reactive Systems* is a remarkable achievement, and fills in book-form a glaring gap in the literature. I heartily recommend it to every serious formal methods theoretician.