Book Reviews

Review of *Concurrency Verification: Introduction to Compositional and Non-compositional Methods*, edited by Willem-Paul de Roever, Frank de Boer, Ulrich Hanneman, Jozef Hooman, Yassine Lakhnech, Mannes Poel and Job Zwiers. Cambridge University Press, 2001. 798 pp. £80. ISBN 0521806089.

As it says in chapter 10, in the 1970s there was a '... golden age of programming when program construction was regarded as an art worthy of gods, and programming concepts were described by mythical terms such as "daemon" and "excorcising"'. Unfortunately the gods proved not to be omnipotent. Bugs that escaped the best minds, such as the famous Woodger and Stenning bug in the concurrent garbage collection algorithm of Dijkstra, Lamport and others, convinced even the most reluctant concurrent algorithm designers that only a fully rigorous state-based proof of correctness (and better still, a machine checked one) could lend any dependability to the business of creating concurrent algorithms. Thus the now enormous field of concurrency verification came into being, and the present book gives a thorough survey of an important part of it, namely techniques stemming from Floyd's inductive assertion method.

This is a fairly large book. It starts with a preface and introductory chapter, by which time we are 80 or so pages in. The preface and introduction summarize, at various levels of detail, the material that is to be treated thoroughly later. Interspersed with this overview are descriptions of historic and contemporary war stories of the pitfalls that lurk in the shadows waiting for the less than thorough concurrent algorithm designer: from the early mutual exclusion exercises of Dijkstra, via the concurrent garbage collection story above, to the more recent Intel Pentium bug and Mars lander disasters. In keeping with the unhurried pace of the book, these stories are mostly told in fair detail, and one can really enjoy them. It is a bitter-sweet irony that through tales like these, the subject of concurrent algorithms makes up for its propensity to humble even the loftiest intellects, by furnishing a rich and readily available fund of anecdotes that enables authors and teachers to spice up their presentations and entice new entrants into this discipline.

The book properly settles into its stride in chapter 2, where Floyd's inductive assertion method is treated in depth. One of the novel features of the book is its careful separation of semantic and syntactic issues in verification theory, a reappraisal of much that is to be found in the literature where many papers pursue what is effectively a mixed or a language directed strategy. The result for Floyd's method is that the essentials are developed for a straightforward transition system model, allowing the results to be adapted to serve as semantic target for later language-based approaches. This lack of syntactic clutter early on makes for a clear and easily comprehensible account.

The next two chapters adapt the bare bones of the Floyd method to the shared variable and synchronous message passing communication models. There are discussions of the methods of Owicki/Gries and of Apt/Francez/de Roever for the two respective cases. As with all the proof methods for concurrency verification discussed in the book, there is detailed and careful coverage of the soundness and semantic completeness of these methods. The semantic completeness approach is characterized by taking the conventional gamut of set theoretic mathematics as given, and ensuring that all needed statements can be conventionally derived.

Completeness itself is critically re-examined in chapter 5. Although many an account of this aspect of verification would be either language based or would content itself with a brief overview of the essential semantic points, the present book gets down to detail and talks about the reduction of semantic completeness to relative completeness of naturally arising languages in the standard model of the naturals (and taking all true facts about the naturals as axioms). Since what is needed later includes being able to speak about arbitrary sequences of values, Gödel encodings are needed, and their construction is carried out from the ground up, after which the relative completeness of the various verification techniques introduced so far can be proved. This material forms the logical bridge between the predominantly semantic treatment of the bulk of the book and the Hoare logic-based parts that occur towards the end.

Then there is the picture gallery. In de Roever and Engelhardt's earlier book on data refinement for sequential systems, the boundary between the theoretical survey and the case studies on specific techniques was marked by a collection of photos of prominent contributors to the subject, flavoured with suitable anecdotes. In this book the photo gallery idea has been retained and expanded, with some enthusiasm. The pictures from the previous book are reproduced in minature, and supplemented by a large collection of additional ones specific to concurrency, with their accompanying stories.

The picture gallery marks the transition between the discussion of noncompositional methods and compositional ones. To give a crude indication of the difference, a noncompositional method will make an assertion X about some state in the transition system of interest, on the assumption that the system is 'all there is'. On the other hand a compositional method will make instead an assertion of the form, *if* W *then* X about the same state, where W captures all the facts that need to be verified by the environment before the system will behave as intended at the given state; on the assumption that there is indeed an environment, and that it is reasonable to require W of it. Thus semantically-based compositional verification techniques form the subject of the next three chapters, with coverage of shared variable and synchronous message passing techniques. This semantic characterization of compositional verification techniques is embellished in the next three chapters to give a discussion of language-based versions of the same things, which is where the Hoare logics mentioned above come into play. At this point one really appreciates the careful distinction between semantic and syntactic methods that has been maintained by the authors throughout. What would otherwise descend into a somewhat amorphous body of mathematics, mixing the two aspects, gets cleanly divided into separate concerns, with appropriate and careful discussion of each on its particular terms.

The last chapter, on communication closed layers, stands a little apart from the rest of the book. No longer concerned purely with the shared variable or synchronous communication techniques, it discusses the transformational design of concurrent algorithms: from a sequence of parallel compositions of small steps, to a parallel composition of sequences of those same small steps. In doing so it has to introduce some major new techniques such as temporal logic, and it feels a little odd to see these things touched on so briefly (being after all topics deserving of whole books in their own right), in stark contrast to the exhaustive depth in which the principal topics of the book are dealt with. And at that point the book ends.

This book is a very substantial and meticulously written treatment of its subject matter, and despite its long author list, a careful and uniform style is maintained throughout. There are briefer treatments available, but none match the attention to detail and generous and complete discussions to be found here. It takes some effort to get through a book of this size, but if you need to learn about concurrency verification in the shared variable and synchronous message passing styles, the effort involved is well worth it. A follow-up volume is promised that illustrates many of the techniques in medium sized applications, accompanied by appropriate supporting material. If this volume is anything to go by, it should be eagerly anticipated.

R. BANACH
University of Manchester

Review of *Neural Networks for Pattern Recognition*, by Christopher M. Bishop. Oxford University Press, 1995. £25. ISBN 0198538499.

Neural networks and pattern recognition are now recognised as two important areas of research in computer science and mathematics. In this book, the author presents fundamental and advanced concepts of artificial neural networks, relating them to problems arising in statistical pattern recognition. This allows for a rigorous and consistent presentation of the ideas underlying neural networks learning and generalisation.

Particularly appealing in this setting is the author's use of radial basis function (RBF) networks as a common ground to present a number of concepts such as function approximation, regularization, and probability density estimation. The author also presents comparative analyses of feedforward networks such as multilayer perceptron and RBF, employing useful graphical illustrations throughout the book. He concludes by introducing Bayesian techniques, providing an early account of the relationships between Bayesian and neural nets learning. An opportunity to provide an early account of support vector machines is missed however. This would be a weakness of any modern neural networks book, particularly if neural networks were to be presented from a statistical learning theory perspective. Also omitted is the family of recurrent neural networks and their learning algorithms.

The author assumes knowledge of elementary linear algebra and of integral and differential calculus. Nevertheless, most of the statistical and mathematical formulations in the book come alive with geometric interpretations that provide a comprehensive understanding of the material. Written in a clear style, *Neural Networks for Pattern Recognition* could be an invaluable reference to either a graduate or an advanced undergraduate student interested in neural networks.

ARTUR S. D'AVILA GARCEZ AND CARLOS E. THOMAZ
City University and Imperial College