The Book Review Column[1]

by William Gasarch
Department of Computer Science
University of Maryland at College Park
College Park, MD, 20742
email: gasarch@cs.umd.edu

Welcome to the Book Reviews Column. We hope to bring you at least two reviews of books every month. In this column four books are reviewed.

1. **Gems of Theoretical Computer Science** by Uwe Schönig and Randall Pruim. Reviewed by: Danny Krizanc. This book is a collection of articles that showcase theorem(s) that the authors think are particularly interesting.
2. **Network Design: Connectivity and Facilities Location, DIMACS Workshop** edited by Pardalos and Du. Reviewed by by Boris Goldengorin. This is a collection of articles on Network Design that were presented at a DIMACS workshop. The intent is to give an up-to-date view of the field.
3. **The Optimal Implementation of Functional Programming Languages** by Andrea Asperti and Stefano Guerrini. Reviewed by Christopher League. This is about issues that arise (and how to resolve them) when trying to implement a functional programming language.
4. **Indiscrete Thoughts** by Gina-Carlo Rota. Reviewed by William Gasarch. This is a collection of essays on math, mathematicians, and related topics. Some of his insights area applicable to Computer Science as well. This review originally appeared in *Journal of Logic and Computation* Vol. 9, No. 4, Aug 1999, pages 595-596. It is reprinted here with permission.

Review of
Gems of Theoretical Computer Science[2]
Authors: Uwe Schöning and Randall Pruim
Publisher: Springer-Verlag, 1998
ISBN 3-540-64425-3
Hardcover
Price: $39.95 (US)

Reviewed by: Danny Krizanc
Wesleyan University, Middletown, Connecticut.

1 Overview

In the summer of 1993, Uwe Schöning taught a course at the Universität Ulm on concepts in theoretical computer science. Rather than taking the standard broad survey approach, Schöning's method was to choose a number of important theorems ("outstanding results") and present them in depth with complete proofs including false leads and important implications. His goal was to have "the students understand and sense 'how theoretical research is done.' " The resulting course materials were later put together and made into a book in German: "Perlen der Theoretischen

Informatik." Randall Pruim has "translated, revised and expanded" that text to give us "Gems of Theoretical Computer Science."[3]

After some introductory material that briefly reviews the most important definitions and results required to understand what follows, the rest of the book consists of 26 chapters, each devoted to a single "gem." The format of each chapter is optimized for achieving Schöning's goal of showing "how theoretical research is done." A chapter usually begins by giving a short informal description of the topic to be considered, how it fits in to theoretical computer science along with motivation for its study. This is followed by any definitions required to formally state the results explored in the chapter. Next comes an exposition of the main theorem and its proof. Unlike standard presentations, the format here includes many exercises that must be solved before continuing. (Solutions to all exercises are provided at the back of the book.) The reader is encouraged to attempt all exercises by interspersing easy with difficult ones and giving insightful hints for the most difficult ones (sometimes all but giving away the answer). Also distinct from standard texts, asides are often followed in depth and a number of consequences of the results are completely worked through. At the end of each chapter we find an annotated set of references for those interested in history, other presentations of the same material and related results.

2 Summary of Contents

The gems Schöning and Pruim have chosen for us are mainly from logic, computability, complexity theory and circuit theory. The topics of the 26 chapters are

1. The priority method and its use in solving "Post's Problem" of whether there exist undecidable, computably enumerable sets that are not Turing equivalent to the halting problem.

2. Hilbert's Tenth Problem: whether there is an algorithmic method for solving Diophantine equations.

3. The equivalence problem for LOOP(1) and LOOP(2) programs.

4. The second LBA problem: whether the class of languages accepted by nondeterministic linear space-bounded Turing machines (LBAs, linear bounded automata) is closed under complement or not.

5. Random walks on graphs, universal traversal sequences and their relation to the study of logarithmic space-bounded Turing machines.

6. An exponential lower bound for the length of resolution proofs of the pigeonhole principle.

7. Characterizing the set of all sizes of finite models of a given formula (the spectral problem) and its relation to the P versus NP question.

8. Applications of Kolmogorov complexity to the universal distribution and average-case complexity.

[3] Not having access to the original and not knowing German in any case, I will not comment on the translation or revisions other than to say that the book does have the "feel" that Schöning suggests he was going for in his preface to the original. As for the expansion, Randall Pruim has added two "gems" of his own (chapters 25 and 26) and has done an admirable job of capturing the same "pleasure of the pursuit of understanding" present in the earlier chapters.

9. Applications of Kolmogorov complexity to lower bounds arguments and the Prime Number Theorem.

10. Probabilistically approximately correct learning and Occam's razor.

11. Lower bounds on the circuit complexity of the parity function using the "random restrictions" method of Furst, Saxe and Sipser.

12. Lower bounds for parity using the algebraic techniques introduced by Razborov and Smolensky.

13. The Craig Interpolation Theorem (for any two formulas F and G in propositional logic such that $F \rightarrow G$ there is a formula H which uses only variables occurring in both formulas such that $F \rightarrow H$ and $H \rightarrow G$) and its relation to the P versus NP question.

14. Testing the equivalence problem of one-time-only branching programs is in *co-RP*.

15. The Berman-Hartmanis Conjecture (all NP-complete languages are pairwise polynomial-time isomorphic to each other) and its relation to sparse NP sets (i.e., Mahaney's theorem).

16. Polynomial-size circuits for NP implies the collapse of the polynomial-time hierarchy.

17. Amplifying probability, recycling random bits and relations between probabilistic complexity classes and the polynomial-time hierarchy.

18. The complement of Graph Isomorphism is in $BP \cdot NP$ (bounded error probability NP) and its implications.

19. Toda's theorem: the polynomial-time hierarchy is contained in $BP \cdot \oplus P$ (bounded error probability Parity P).

20. Introduction to interactive proof systems and the class IP, including a zero-knowledge protocol for Graph Isomorphism.

21. $IP = PSPACE$.

22. $P^A \neq NP^A$ with probability one for a randomly chosen oracle A.

23. The construction of linear size superconcentrators.

24. Pebble games and the relationship between deterministic time and space.

25. Levin's theory of average-case complexity.

26. Grover's quantum algorithm for searching a database.

3 Opinion

At times the book shows some of the weaknesses of having its source in course materials (e.g., some repetitions, inconsistent notation, typographical errors) but these minor inconveniences are easily overlooked. They are far outweighed by the strongest feature of the book: its format. It is clear that a lot of work has gone into choosing just the right exercises to expose the inner workings of each proof. This makes it ideal for self-study by graduate students (or talented undergraduates) new to the material or experienced researchers looking to expand their understanding of something they were aware of but had never studied in depth. As somone who works mainly in algorithms, most of these topics were either new to me or I hadn't considered them since leaving graduate school. As such, I found the book to be a perfect oppurtunity to pick up topics I had missed or to refresh my memory of things long forgotten. While it is always possible to quibble with the choice of topics (everyone will have at least one or two favorite results left out and one person's gem is often a piece of coal to another), the book would make a good text for an advanced graduate course on computability and complexity for students interested in pursuing research in these areas. An interesting project for such a course might be to have the students develop their own chapters from a gem of their choosing. With any luck this book will inspire imitations in other areas of computer science.

Review of

Network Design: Connectivity and Facilities Location
(Proceedings from DIMACS workshop in April 1997
Publisher: AMS 1998
ISBN 0821808346
$79.50, Hardcover

Review by Boris Goldengorin

Positive impressions:

1. *The title of this book and included papers*: all papers really correspond to the title of this book except the paper "A case Study of De-randomization Methods for Combinatorial Optimization" by José D.P. Rolin and Luca Trevisan. Anyway, this paper reflects one of the general topics in "Discrete Mathematics" and devoted to the analysis of the *conditional probabilities method* for routing, packing and covering integer linear programming problems. It is well known that network design problems can be formulated as an integer programming problem, the relaxation over the reals of which is usually used to develop approximations algorithms. For an integer linear programming formulation we usually construct a pair of feasible solutions (primal and dual relaxation) and take the ratio between the values of these primal and dual solutions as a constant r for the corresponding r-approximation algorithm. For an integer mathematical programming formulation we use a pair of the relaxed and rounded solution. If the ratio of the values of these rounded and relaxed solutions is bounded by a constant r then it tells us that the rounded solution is r-approximate. In the second case rounding a fractional solution is a difficult task. Authors have used *randomized rounding method* and have proved their approximation guarantee by using Chernoff bounds. By using the so called *pessimistic estimators* authors de-randomize their algorithm. Both of above mentioned methods can be considered as authors' contribution in theoretical computer science.

2. In papers by Siu-Wing Cheg; Dietmar Cieslik; Cees Duin; Andreas Eisenblätter; Thomas Erlebach, Klaus Jansen, Christos Kaklamanis, and Pino Persiano; Krisina Holmqvist, Athanasios Migdalas, and Panos M. Pardalos; Klaus Jansen; Sudipto Guha and Samir Khuller; Wu-Ji and J. Macgregor Smith the potential applications in Computer Science are discussed.

3. Very interesting result is announced in the paper by Marek Karpinsky and Alexander Zelikovsky. They show that the so called *ε-dense set cover problem* can be approximated with the performance ratio $c \log k$ for any $c > 0$ though it is unlikely to be NP-hard. This is in contrast to the hardness results for set cover proven by Lund and Yannakakis (1994) and Feige (1996) that the set cover problem cannot be approximated to within factor $(1 - o(1)) \ln k$ un-