Theoretical Aspects of Local Search
by Wil P. A. J. Michiels, Emile H. L. Aarts, and Jan H. M. Korst,
published by Springer in the EATCS Series Monographs in Theoretical Computer Science, 2007
(ISBN: 978-3-540-35853-4, hardcover, 235 pages, EUR 54.95)
reviewed by Jakub Mareček

"Theoretical aspects of local search? That must be like two pages long!" said a seasoned theoretical computer scientist, when he learned what a book was under review. Many computer scientists do indeed share a scathing contempt for anything heuristic. Perhaps this is because they do not realise that many real-life optimisation problems do boil down to large instances of hard-to-approximate problems, well beyond the reach of any exact method known today. Perhaps they do not realise that performance of exact solvers for hard problems is largely dependent on the quality of in-built heuristics. Certainly, this leads to the irrelevance of a large body of work in Theoretical Computer Science to the operations research industry, and a certain lack of analytical approach in real-life problem-solving, to say the least. Any attempt to change this situation would be most welcome. Local search, readily admitting theoretical analysis, might be a good starting point.

Design, analysis, and empirical evaluation of algorithms for hard problems generally involve a number of interesting challenges. No matter whether you are designing integer programming solvers, model checkers, or flight scheduling systems, you have to use heuristics, which present-day Theoretical Computer Science by and large ignores. For instance in precoloured bounded/equitable graph colouring, which underlies many timetabling and scheduling problems, the present-best theoretical result says it is $\mathcal{NP}$-Complete for trees and hence a tree-width decomposition doesn't help (Bodlaender & Fomin, 2005), in addition to being hard to approximate in general (Zuckerman, 2007). Despite the depth of these negative results, their practical utility might be somewhat hard to see for a person working on solvers for the very problem. The divorce of theoretical study of approximability of rather obscure problems and special cases from the development of solvers for messy real-life problems of operations research, is also reflected in the in the usual undergraduate Computer Science curriculum. If any attention is given to heuristics (see Kleinberg & Tardos, 2005, for a rare example), the treatment is often confined to the plain old greedy "iterated improvement" and some elementary stochastic local search heuristics, which may seem too trivial even to an undergraduate. The importance of the choice of neighbourhood and objective function is often stressed, but rarely demonstrated. An abstract discussion of the trade-off between the search across many basins of attraction within the search space (diversification) and the convergence to local optima within each basin (intensification) sometimes follows. (See the gripping personal account of Glover (2007).) Only few textbooks proceed to mention heuristic pre-processing and decompositions, or the role of pre-processing within heuristics (at least, for instance, sorting with tie-breaking). Even less attention is usually focused on the concept of relaxation and restriction, such as forbidding features detected in many bad solutions (Steiglitz & Weiner, 1968), or fixing features common to many (Lin, 1965), instance analysis determining which algorithm to use, or auto-tuning (Glover & Greenberg, 1989), vital to industrial solvers. As non-trivial analyses of expected behaviour remain rare, the treatment is often concluded with a description of the worst case scenario, where the heuristic takes infinitely long to converge or does not converge at all. Not only research, but also the usual treatment in the curriculum tend to remain rather distant to real-world solver design and seem best read with a funeral march playing in the background.

The authors of the book under review are in an excellent position to write a very different account. On one hand, at least two of them are well-known in the algorithm design community, not least for their grim analyses of local search heuristics (see the conclusions of Aarts, Korst, and Zwietering (1996): "simulated annealing algorithm is clearly unsuited for solving combinatorial optimization problems [to optimality]") and vocal contempt for "home-made voodoo optimisation techniques inspired by some sort of physical or biological analogy rather than by familiarity with what has been achieved in mathematics" (Aarts & Lenstra, 1997b). On the other hand, however, all three authors are involved in the design of heuristics solving real-life problems at Philips Research. It might then seem reasonable to expect the book under review to be more upbeat than the usual dead march.

The book under review is clearly aimed at advanced undergraduate students, although the blurb mentions "researchers and graduate students" as the intended audience. At 185 pages bar the appendices, the book is of the right length for a relaxed, largely self-contained term-long course. In chapter two, it introduces five problems used as running examples throughout the book (TSP, vertex colouring, Steiner trees, graph partitioning and make-span scheduling). Chapters 3 and 4 introduce the concept of a neighbourhood and discuss properties of various neighbourhoods for the example problems. Chapter 5 introduces rudiments of approximability and hardness of approximation and presents elementary proofs of performance guarantees for some of the local search methods introduced previously. Chapter 6 recalls the existence of some non-trivial results on the complexity of local search. Chapter 7 previews some general heuristic design schemes ("metaheuristics"), such as simulated annealing. Chapter 8 introduces Markov chains, conditions of their ergodicity and convergence properties of simulated annealing. Finally, three appendices include a list of 12 problems-neighbourhood pairs, which are known to be $\mathcal{PLS}$-complete (see below). All chapters are accompanied by exercises and most have the very comforting feel of an easily understandable lecture transcribed, while maintaining a reasonable level of rigour.

For a narrowly-focused researcher in Complexity, only Chapter 6 may be of immediate interest. It provides a concise overview of $\mathcal{PLS}$-Completeness, a framework for reasoning about complexity of local search algorithms and an alternative measure of complexity of "easier hard" problems, introduced by Johnson, Papadimtriou, and Yannakakis (1988). A pair of a given problem and neighbourhood structure falls into the class of $\mathcal{PLS}$ (poly-time searchable), if an initial solution can be obtained using poly-time, there is a poly-time evaluation function, and a poly-time test of optimality within a neighbourhood, producing an improving solution together with the negative answer. From this follow the concepts of $\mathcal{PLS}$-Reducibility and $\mathcal{PLS}$-Completeness. Unless $\mathcal{NP} =$ co-$\mathcal{NP}$, problems in $\mathcal{PLS}$ are not $\mathcal{NP}$-Hard. For certain problems, the choice of neighbourhood seems relatively restricted and results suggesting the impossibility of finding an improving solution fast in a commonly used neighbourhood might seem damning. Unlike in approximability, results in $\mathcal{PLS}$-Completeness are, however, tied to a particular neighbourhood, which is similar to results in parametrised complexity being tied to a particular decomposition. Introduction of a new neighbourhood structure or decomposition (Hliněný, Oum, Seese, & Gottlob, to appear) may bring new breakthroughs. These connections and ramifications are, however, not discussed in the book. At 36 pages, the chapter on $\mathcal{PLS}$-Completeness is more concise than the paper that has introduced the concept (Johnson et al., 1988) or the exposition by Yannakakis (1997) "it is based on" (p. 98, sic!), and does not seem to represent good value for money if the rest of the book is of little or no interest to the reader.

Opinion: Overall, the book provides a concise and easily understandable introduction to the basics of local search, an important concept in the design of heuristics. As it gracefully omits anything and everything a *theoretical* computer scientist need not learn from the arguably bloated jargon of heuristic design as well as applications to complex real-life problems, it is well-suited for a term-long course on heuristic design for theoretically-inclined undergraduates and first-year graduate students. In other circumstances, either a chapter in a good undergraduate algorithm design textbook (Kleinberg & Tardos, 2005) with some project-work, a broader survey of heuristic and exact optimisation (Burke & Kendall, 2005), or a more comprehensive study on local search (Aarts & Lenstra, 1997a; Hoos & Stützle, 2004) might be worth consideration.

References

Aarts, E. H. L., Korst, J. H. M., & Zwietering, P. (1996). Deterministic and randomized local search. In P. Smolensky, M. C. Mozer, & D. E. Rumelhart (Eds.), *Mathematical perspectives of neural networks* (pp. 143–224). Mahwah, NJ: Lawrence Erlbaum Associates.

Aarts, E. H. L., & Lenstra, J. K. (1997a). *Local search in combinatorial optimization*. Chichester, UK: Wiley-Interscience.

Aarts, E. H. L., & Lenstra, J. K. (1997b). Preface. In *Local search in combinatorial optimization* (p. vii-viii). Chichester, UK: Wiley-Interscience.

Bodlaender, H. L., & Fomin, F. V. (2005). Equitable colorings of bounded treewidth graphs. *Theoret. Comput. Sci.*, *349*(1), 22–30.

Burke, E. K., & Kendall, G. (Eds.). (2005). *Search methodologies: Introductory tutorials in optimization and decision support techniques*. New York, NY: Springer.

Glover, F. (2007). Tabu search – uncharted domains. *Ann. Oper. Res.*, *149*, 89–98.

Glover, F., & Greenberg, H. J. (1989). New approaches for heuristic search: A bilateral linkage with artificial intelligence. *European J. Oper. Res.*, *39*(2), 119–130.

Hliněný, P., Oum, S. il, Seese, D., & Gottlob, G. (to appear). Width parameters beyond tree-width and their applications. *Computer J.*

Hoos, H., & Stützle, T. (2004). *Stochastic local search: Foundations & applications*. San Francisco, CA: Morgan Kaufmann.

Johnson, D. S., Papadimtriou, C. H., & Yannakakis, M. (1988). How easy is local search? *J. Comput. Syst. Sci.*, *37*(1), 79–100.

Kleinberg, J., & Tardos, E. (2005). *Algorithm design*. Boston, MA: Addison-Wesley Longman.

Lin, S. (1965). Computer solutions to the travelling salesman problems. *Bell Syst. Tech. J.*, *44*(10), 2245–2269.

Steiglitz, K., & Weiner, P. (1968). Some improved algorithms for computer solution of the traveling salesman problem. In *Proc. 6th annual Allerton conference on circuit and systems theory* (pp. 814–821). Urbana, IL.

Yannakakis, M. (1997). Computational complexity. In *Local search in combinatorial optimization* (pp. 19–55). Chichester, UK: Wiley-Interscience.

Zuckerman, D. (2007). Linear degree extractors and the inapproximability of max clique and chromatic number. *Theory Comput.*, *3*(6), 103–128.