AI Magazine Volume 25 Number 4 (2004) (© AAAI)

Principles of Constraint Programming and *Constraint Processing*: A Review

Peter van Beek and Toby Walsh

■ Apt, Krzysztof. *Principles of Constraint Programming*. Cambridge, England: Cambridge University Press. ISBN: 0-521-825830. 420 pages, $50.00. Publication Date: August 2003. http://uk.cambridge.org

■ Dechter, Rina. *Constraint Processing*. San Francisco: Morgan Kaufmann Publishers. ISBN 1-55860-890-7. 481 pages, $65.95. Publication Date: May 2003. http://www.mkp.com

Texts in constraint programming are a little like buses. You wait forever for one to come along, and then two come along at once. In this case, there has been a large gap in the market for a theoretical introduction to constraint programming ever since Edward Tsang's *Foundations of Constraint Satisfaction* (1993) went out of print.[1] Therefore, we are very pleased to see two books written by two of the leading researchers in this field come along to fill the gap.

Constraint programming is a very active research area within AI. It is a highly successful technology for solving a wide range of combinatorial problems, including scheduling, rostering, assignment, routing, and design. A number of companies, like ILOG, Dash Optimization, and Parc Technologies, market model building and constraint programming toolkits, which are used by companies as diverse as Amazon.com, British Airways, Chevron, Cisco, Dupont, Ford, General Mills, HP, I2, JD Edwards, KLM, Lockheed Martin, Mannersmann, Nestle, Oracle, Proctor & Gamble, Qwest, Renault, SNCF, UPS, and Volvo.

Constraint programming is a declarative style of modeling combinatorial problems in which the user identifies the decision variables, their possible domain of values, and specifies constraints over the allowed values (for example, no two of these variables can take the same value). Sophisticated but general purpose AI search techniques like constraint propagation (to prune irrelevant parts of the search tree) and dependency directed backtracking can then be used to find solutions.

Given the many advances made in constraint programming over the last decade, a new text would have been needed even if Edward Tsang's book had remained in print. These two new texts are written by two of the leading researchers in this field. *Principles of Constraint Programming* by Krzysztof Apt contains chapters that cover topics like local consistency, constraint propagation, linear equations, interval reasoning, and search. *Constraint Processing* by Rina Dechter covers similar ground but also has chapters that cover topics like local search, tree decomposition methods, optimization, and probabilistic networks more extensively. Dechter's book also contains a chapter by David Cohen and Peter Jeavons on tractability and one by Francesca Rossi on constraint logic programming.

Spot the Similarities

There is much in common between the two books. This is perhaps not so surprising since Krzysztof Apt thanks Rina Dechter for much useful discussion that helped him enter the field and start doing research in the area. Both texts cover the basics of con-

straint programming, such as local consistency and backtracking search methods, as well as more specialized topics like linear equations and temporal constraints. Both texts could be used as part of a graduate course on constraint programming. In both books, new concepts are illustrated with many helpful examples. In addition, both texts end each chapter with a useful bibliographical section for further reading, as well as exercises for the enthusiastic student.

Spot the Differences

From a stylistic perspective, the two books are very different. Apt's book is a thorough introduction to constraint propagation and search and would appeal to readers who want to study the topic on their own. Indeed, the book is very much a description of Apt's own voyage discovering these essentials of constraint programming. Dechter's book, on the other hand, is the broader of the two texts and perhaps more suited as a text to accompany a broader-based graduate-level course on the topic.

From a theoretical perspective, the two books are also very different. For example, consider their treatment of constraint propagation algorithms, a topic that goes to the core of constraint programming. In Dechter's book, constraint propagation algorithms are presented via queue-based algorithms like AC-3. In Apt's book, on the other hand, constraint propagation algorithms are presented via the generic iteration framework in which we seek the fixed point of a set of rules. Both perspectives have their advantages and disadvantages, so it is a little disappointing that you need to read both books to see them both.

Mind the Gaps

If you were being fussy, you might also express disappointment that a number of basic topics are not covered by either of the texts. For example, neither book covers in significant depth modeling, global constraints, symmetry, and extended case studies of applications. Constraint programming is now a large and very active research area. A single volume is by necessity going to have to have to skip certain topics. Beyond the constraint programming books mentioned so far in this review—where the approach is a theoretical, language-independent introduction to the field—there is an alternative stream of texts whose approach is grounded in constraint logic programming (Fruhwirth and Abdennadher 2003; Marriott and Stuckey 1998; Van Hentenryck 1989). Fortunately, some of these basic topics are well-covered in these general books from the constraint logic programming perspective. As well, there are now appearing book-length treatments of modeling and global constraints in important application areas (such as Baptiste, Le Pape, and Nuijten 2001).

Conclusion

We can recommend that you read both of these excellent books, in part, because they approach the subject from the authors' different perspectives and each of them brings something useful to the table. Another reason for reading both is that the topics covered by the two texts do not overlap completely, and the union of their contents is a better introduction to constraint programming. If you only want to read one book, Apt's may appeal more to the reader who likes a gentle self-guided introduction or the lecturer who wants a text to accompany a course more focused on constraint propagation and search. On the other hand, Dechter's may appeal more to the reader who wants a reference work to dip into or the lecturer who wants a text to accompany parts of a graduate course whose aim is to survey more of the field of constraint programming.

Note

1. Reprints are available from the author.

References

Baptiste, P.; Le Pape, C.; and Nuijten, W. 2001. *Constraint-Based Scheduling: Applying Constraint Programming to Scheduling Problems*. Dortrecht, The Netherlands: Kluwer Academic Publishers.

Fruhwirth, T.; and Abdennadher, S. 2003. *Essentials of Constraint Programming*. Berlin: Springer.

Marriott, K.; and Stuckey, P. J. 1998. *Programming with Constraints*. Cambridge, Mass.: The MIT Press.

Tsang, E. P. K. 1993. *Foundations of Constraint Satisfaction*. Boston: Academic Press.

Van Hentenryck, P. 1989. Constraint Satisfaction in Logic Programming. Cambridge, Mass.: The MIT Press.

Peter van Beek (vanbeek@uwaterloo.ca) received his Ph.D. in 1990 from the University of Waterloo and then joined the faculty at the University of Alberta. Ten years later he returned to Waterloo. He is soon to take over as editor of the journal *Constraints*.

Toby Walsh (tw@cse.unsw.edu.au) recently moved to National Information and Communications Technology Australia (NICTA) and the University of New South Wales in Sydney, Australia. Previously he was an SFI Research Professor at University College Cork. He is also an external professor of Uppsala University. He is an ECCAI Fellow and editor of *AI Communications*.