

Essentials Of Constraint Programming 1st Edition

*Handbook of Knowledge Representation describes the essential foundations of Knowledge Representation, which lies at the core of Artificial Intelligence (AI). The book provides an up-to-date review of twenty-five key topics in knowledge representation, written by the leaders of each field. It includes a tutorial background and cutting-edge developments, as well as applications of Knowledge Representation in a variety of AI systems. This handbook is organized into three parts. Part I deals with general methods in Knowledge Representation and reasoning and covers such topics as classical logic in Knowledge Representation; satisfiability solvers; description logics; constraint programming; conceptual graphs; nonmonotonic reasoning; model-based problem solving; and Bayesian networks. Part II focuses on classes of knowledge and specialized representations, with chapters on temporal representation and reasoning; spatial and physical reasoning; reasoning about knowledge and belief; temporal action logics; and nonmonotonic causal logic. Part III discusses Knowledge Representation in applications such as question answering; the semantic web; automated planning; cognitive robotics; multi-agent systems; and knowledge engineering. This book is an essential resource for graduate students, researchers, and practitioners in knowledge representation and AI. * Make your computer smarter * Handle qualitative and uncertain information * Improve computational tractability to solve your problems easily*

Constraint programming aims at supporting a wide range of complex applications, which are often modeled naturally in terms of constraints. Early work, in the 1960s and 1970s, made use of constraints in computer graphics, user interfaces, and artificial intelligence. Such work introduced a declarative component in otherwise-procedural systems to reduce the development effort.

Constraint programming (CP) is a declarative programming paradigm with many academic and industrial applications (from n-queens to planning, vehicle routing, and optimization, among other fields). Music composition has been one of these applications since the earliest works on automatic harmonization, and it remains a very special and challenging one due to its artistic (and highly subjective) nature. The early works on CP in music were limited to classical music composition, as the harmonization and counterpoint rules naturally translate into constraints. However, when contemporary composers began to be interested in constraints, CP became an essential tool in computer-assisted composition systems. As several contemporary musical pieces have now been composed "with constraints", it is reasonable to ask why CP applies so naturally to music, and what the particular features of musical problems are. This book presents information about recently developed musical CP systems from both the scientist's and composer's point-of-view. It will therefore be of interest to students and researchers of music technology, composers in the computer music scene, and music software companies—especially those trying to model high level musical behaviors (i.e., intelligent arpeggiation/arrangement on synthesizers, "Band in a Box" software, etc.), perform music data mining, and execute music taste engineering for online music delivery.

Authored by two of the leading authorities in the field, this guide offers readers the knowledge and skills needed to achieve proficiency with embedded software.

On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE

Constraint Processing

Principles and Practice of Constraint Programming - CP 2005

9th Asian Computing Science Conference. Dedicated to Jean-Louis Lassez on the Occasion of His 5th Cycle Birthday, Chiang Mai, Thailand, December 8-10, 2004

Handbook of Constraint Programming

In many application areas, it is necessary to make effective decisions under constraints. Several area-specific techniques are known for such decision problems; however, because these techniques are area-specific, it is not easy to apply each technique to other applications areas. Cross-fertilization between different application areas is one of the main objectives of the annual International Workshops on Constraint Programming and Decision Making. Those workshops, held in the US (El Paso, Texas), in Europe (Lyon, France) and in Asia (Novosibirsk, Russia), from 2008 to 2012, have attracted researchers and practitioners from all over the world. This volume presents extended versions of selected papers from those workshops. These papers deal with all stages of decision making under constraints: (1) formulating the problem of multi-criteria decision making in precise terms, (2) determining when the corresponding decision problem is algorithmically solvable; (3) finding the corresponding algorithms and making these algorithms as efficient as possible and (4) taking into account interval, probabilistic and fuzzy uncertainty inherent in the corresponding decision making problems. The resulting application areas include environmental studies (selecting the best location for a meteorological tower), biology (selecting the most probable evolution history of a species), and engineering (designing the best control for a magnetic levitation train).

This book constitutes the refereed proceedings of the 8th International Conference on Artificial Intelligence and Symbolic Computation, AISC 2006, held in Beijing, China in September 2006. The 18 revised full papers presented together with 4 invited papers were carefully reviewed and selected from 39 submissions. Based on heuristics and mathematical algorithmics, artificial intelligence and symbolic computation are two views and approaches for automating (mathematical) problem solving. The papers

address all current aspects in the area of symbolic computing and AI: mathematical foundations, implementations, and applications in industry and academia. The papers are organized in topical sections on artificial intelligence and theorem proving, symbolic computation, constraint satisfaction/solving, and mathematical knowledge management.

A special mention for 2004 is in order for the new Doctoral Symposium Workshop where three young postdoc researchers organized an original setup and formula to bring PhD students together and allow them to submit their research proposals for selection. A limited number of the submissions and their approaches were independently evaluated by a panel of senior experts at the conference, and presented by the students in front of a wider audience. These students also got free access to all other parts of the OTM program, and only paid a heavily discounted fee for the Doctoral Symposium itself. (In fact their attendance was largely sponsored by the other participants!) If evaluated as successful, it is the intention of the General Chairs to expand this model in future editions of the OTM conferences and so draw in an audience of young researchers to the OnTheMove forum. All three main conferences and the associated workshops share the distributed aspects of modern computing systems, and the resulting applications - pull created by the Internet and the so-called Semantic Web. For DOA 2004, the primary emphasis stayed on the distributed object infrastructure; for ODBASE 2004, it was the knowledge bases and methods required for enabling the use of formal semantics; and for CoopIS 2004 the main topic was the interaction of such technologies and methods with management issues, such as occurs in networked organizations. These subject areas naturally overlap and many submissions in fact also treat envisaged mutual impacts among them.

The use of constraints had its scientific and commercial breakthrough in the 1990s. Programming with constraints makes it possible to model and specify problems with uncertain, incomplete information and to solve combinatorial problems, as they are abundant in industry and commerce, such as scheduling, planning, transportation, resource allocation, layout, design, and analysis. This book is a short, concise, and complete presentation of constraint programming and reasoning, covering theoretical foundations, algorithms, implementations, examples, and applications. It is based on more than a decade of experience in teaching and research about this subject. This book is intended primarily for graduate students, researchers, and practitioners in diverse areas of computer science and related fields, including programming languages, computational logic, symbolic computation, and artificial intelligence. The book is complemented by a web-page with teaching material, software, links, and more. We take the reader on a step-by-step journey through the world of constraint-based programming and constraint reasoning. Feel free to join in ... Acknowledgements Thorn thanks his wife Andrea and his daughter Anna - for everything. He dedicates his contribution to the book to the memory of his mother, Grete. Slim thanks his wife Nabila and his daughters Shirine and Amira for their ongoing support and patience.

Logic-Based Program Synthesis and Transformation

Principles and Practice of Constraint Programming

17th International Symposium, LOPSTR 2007, Kongens Lyngby, Denmark, August 23-24, 2007, Revised Selected Papers

Convex Optimization

With C and GNU Development Tools

Toward a Unified Methodology

Constraint Programming is a problem-solving paradigm that establishes a clear distinction between two pivotal aspects of a problem: (1) definition of the constraints that define the problem to be solved and (2) the algorithms and heuristics enabling the selection of decision problem. It is because of these capabilities that Constraint Programming is increasingly being employed as a problem-solving tool to solve problems. Hence the development of Constraint-Based Scheduling as a field of study. The aim of this book is to provide an overview of Constraint-Based Scheduling techniques. Following the principles of Constraint Programming, the book consists of three distinct parts: introduces the basic principles of Constraint Programming and provides a model of the constraints that are the most often encountered problems. Chapters 2, 3, 4, and 5 are focused on the propagation of resource constraints, which usually are responsible for the "hard" scheduling problem. Chapters 6, 7, and 8 are dedicated to the resolution of several scheduling problems. These examples illustrate the practical efficiency of the constraint propagation methods of the previous chapters. They also show that besides constraint propagation the search space must be carefully designed, taking into account specific properties of the considered problem (e.g., dominance relations, possible use of decomposition rules). Chapter 9 mentions various extensions of the model and presents promising research directions. Concurrent Constraint Programming introduces a new and rich class of programming languages based on the notion of computing with information, or constraints, that synthesize and extend work on concurrent logic programming and that offer a promising approach for in the semantics of concurrent, nondeterministic programming languages. Saraswat develops an elegant and semantically tractable framework for computing with constraints, emphasizing their importance for communication and control in concurrent, programming languages. He describes a paradigm, illustrates its structure, discusses various augmentations, gives a simple implementation of a concrete language, and specifies it with other formalisms. In this framework, concurrently executing agents communicate by placing and checking constraints on shared variables. The major form of concurrency control in the system is through the operations of Atomic Tell - an agent may instantaneously place constraints that are consistent with constraints that have already been placed - and Blocking Ask - an agent must block when it checks a constraint to hold. Other operations at a finer granularity of atomicity are also presented. Saraswat introduces and develops the concurrent constraint programming languages based on these ideas, shows how various constraint systems can naturally realize data structures common in AI, and presents a formal operational semantics for many languages in the concurrent constraint family. In addition, he provides a concrete paradigm on a sequential machine by presenting a compiler for the concurrent constraint language Herbrand and demonstrates a number of concurrent programming techniques that lead to novel presentations of algorithms for many concurrent programming problems. V. Member of the Research Staff at Xerox Palo Alto Research Center.

This book introduces a new logic-based multi-paradigm programming language that integrates logic programming, functional programming, programming with tabling, and scripting, for use in solving combinatorial search problems, including CP, SAT, and MIP (mixed integer programming) based solver modules, and a module for planning that is implemented using tabling. The book is useful for undergraduate and graduate students, researchers, and practitioners.

The Springer Handbook for Computational Intelligence is the first book covering the basics, the state-of-the-art and important applications in this rapidly expanding discipline of computational intelligence. This comprehensive handbook makes readers familiar with a broad spectrum

to solve various problems in science and technology. Possible approaches include, for example, those being inspired by biology, living organisms, and animate systems. Content is organized in seven parts: foundations; fuzzy logic; rough sets; evolutionary computation; neural networks; and hybrid computational intelligence systems. Each Part is supervised by its own Part Editor(s) so that high-quality content as well as assurance is assured.

Advances in Computer Science - ASIAN 2004, Higher Level Decision Making

The Newport Papers

8th International Conference, AISC 2006, Beijing, China, September 20-22, 2006, Proceedings

Programming Embedded Systems

Recent Advances in Constraints

17th International Conference, CPAIOR 2020, Vienna, Austria, September 21-24, 2020, Proceedings

Evolutionary scheduling is a vital research domain at the interface of artificial intelligence and operational research. This edited book gives an overview of many of the current developments in the large and growing field of evolutionary scheduling. It demonstrates the applicability of evolutionary computational techniques to solve scheduling problems, not only to small-scale test problems, but also fully-fledged real-world problems.

Constraint satisfaction is a simple but powerful tool. Constraints identify the impossible and reduce the realm of possibilities to effectively focus on the possible, allowing for a natural declarative formulation of what must be satisfied, without expressing how. The field of constraint reasoning has matured over the last three decades with contributions from a diverse community of researchers in artificial intelligence, databases and programming languages, operations research, management science, and applied mathematics. Today, constraint problems are used to model cognitive tasks in vision, language comprehension, default reasoning, diagnosis, scheduling, temporal and spatial reasoning. In Constraint Processing, Rina Dechter, synthesizes these contributions, along with her own significant work, to provide the first comprehensive examination of the theory that underlies constraint processing algorithms. Throughout, she focuses on fundamental tools and principles, emphasizing the representation and analysis of algorithms. Examines the basic practical aspects of each topic and then tackles more advanced issues, including current research challenges. Builds the reader's understanding with definitions, examples, theory, algorithms and complexity analysis. Synthesizes three decades of researchers work on constraint processing in AI, databases and programming languages, operations research, management science, and applied mathematics

This book constitutes the refereed proceedings of the First International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems, CPAIOR 2004, held in Nice, France in April 2004. The 23 revised full papers and 7 revised short papers presented together with an invited talk were carefully reviewed and selected from 56 submissions. Methodological and foundational issues from AI, OR, and algorithmics are presented as well as applications to the solution of combinatorial optimization problems in various fields via constraint programming.

This book covers recent progress in solving propositional satisfiability and related problems. Propositional satisfiability is a powerful and general formalism used to solve a wide range of important problems including hardware and software verification. Research into methods to automate such reasoning has therefore a long history in artificial intelligence. This book follows on from the highly successful volume entitled SAT 2000 published five years ago.

Evolutionary Scheduling

Joint ERCIM/CoLogNET International Workshop on Constraint Solving and Constraint Logic Programming, CSCLP 2005, Uppsala, Sweden, June 20-22, 2005, Revised Selected and Invited Papers

OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2004, Agia Napa, Cyprus, October 25-29, 2004. Proceedings. Part II

Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems

Handbook of Knowledge Representation

Integration of Constraint Programming, Artificial Intelligence, and Operations Research

A comprehensive introduction to the tools, techniques and applications of convex optimization.

Mechanical Engineering domain problems are generally complex, consisting of different design variables and constraints. These problems may not be solved using gradient-based optimization techniques. The stochastic nature-inspired optimization techniques have been proposed in this book to efficiently handle the complex problems. The nature-inspired algorithms are classified as bio-inspired, swarm, and physics/chemical-based algorithms. Socio-inspired is one of the subdomains of bio-inspired algorithms, and Cohort Intelligence (CI) models the social tendencies of learning candidates with an inherent goal to achieve the best possible position. In this book, CI is investigated by solving ten discrete variable truss structural problems, eleven mixed variable design engineering problems, seventeen linear and nonlinear constrained test problems and two real-world applications from manufacturing domain. Static Penalty Function (SPF) is also adopted to handle the linear and nonlinear constraints, and limitations in CI and SPF approaches are examined. Constraint Handling in Cohort Intelligence Algorithm is a valuable reference to practitioners working in the industry as well as to students and researchers in the area of optimization methods.

This volume contains the papers presented at the Eighth International Symposium on Practical Aspects of Declarative Languages (PADL 2006) held on January 9-10, 2006, in Charleston, South Carolina. Information about the conference can be found at <http://www.cs.brown.edu/people/pvh/PADL06.html>. As is now traditional, PADL 2006 was co-located with the 33rd Annual Symposium on Principles of Programming Languages that was held on January 11-13, 2006. The PADL conference series is a forum for researchers and practitioners to present original work emphasizing novel applications and implementation techniques for all forms of declarative concepts. Topics of interest include, but are not limited to: – Innovative applications of declarative languages; – Declarative domain-specific languages and applications; – Practical applications of theoretical results; – New language developments and their impact on applications; – Evaluation of implementation techniques on practical applications; – Novel implementation techniques relevant to applications; – Novel uses of declarative languages in the classroom; – Practical experiences. This year, there were 36 submissions. Each submission was reviewed by at least three Programme Committee members. The committee decided to accept 15 papers. In addition, the programme also included three invited talks by Erik Meijer,

David Roundy, and Philip Walder.

This book constitutes the thoroughly refereed post-conference proceedings of the 29th International Symposium on Logic-Based Program Synthesis and Transformation, LOPSTR 2019, held in Porto, Portugal, in October 2019. The 15 revised full papers were carefully reviewed and selected from 32 submissions. In addition to the 15 papers, this volume includes 2 invited papers. The symposium cover all aspects of logic-based program development, stages of the software life cycle, and issues of both programming-in-the-small and programming-in-the-large. This year LOPSTR extends its traditional topics to include also logic-based program development based on integration of sub-symbolic and symbolic models, on machine learning techniques and on differential semantics. The papers are grouped into the following topics: static analysis, program synthesis, constraints and unification, debugging and verification, and program transformation.

Essentials of Programming Languages

Artificial Intelligence and Symbolic Computation

Springer Handbook of Computational Intelligence

Proceedings of the ... International ACM SIGPLAN Conference on Principles and Practice of Declarative Programming

Constraint and Integer Programming

Constraint-Based Scheduling

This textbook offers an understanding of the essential concepts of programming languages. The text uses interpreters, written to express the semantics of many essential language elements in a way that is both clear and directly executable.

The 11th International Conference on the Principles and Practice of Constraint Programming (CP 2005) was held in Sitges (Barcelona), Spain, October 1-5, 2005. Information about the conference can be found on the web at

<http://www.iiia.csic.es/cp2005/>. Information about past conferences in the series can be found at <http://www.cs.ualberta.ca/~ai/>

The conference series is the premier international conference on constraint programming and is held annually. The conference is concerned with all aspects of computing with constraints, including: algorithms, applications, environments, languages, models and systems. This year received 164 submissions. All of the submitted papers received at least three reviews, and the papers and their reviews were then discussed during an online Program Committee meeting. As a result, the Program Committee chose 48 (29.3%) papers to be published in the proceedings and a further 22 (13.4%) papers to be published as short papers. The full papers were presented at the conference in two parallel tracks and the short papers were presented as posters during a lively evening session. Two papers were selected by a subcommittee of the Program Committee--consisting of Chris Beck, Gilles Pesant, and myself--to receive best paper awards. The conference also included excellent invited talks by Hector Geffner, Ian Horrocks, Francesca Rossi, and Peter J. Stuckey. As a permanent record, the proceedings contain four-page extended abstracts of the invited talks.

The volume LNCS 12296 constitutes the papers of the 17th International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research which will be held online in September 2020. The 32 regular papers presented together with 4 abstracts of fast-track papers were carefully reviewed and selected from a total of 72 submissions. Additionally, this volume includes the 4 abstracts and 2 invited papers by plenary speakers. The conference program also included a Master Class on the topic "Advances in Optimization Paradigms and Solving Technology"

Answer set programming (ASP) is a programming methodology oriented towards combinatorial search problems. In such a problem, the goal is to find a solution among a large but finite number of possibilities. The idea of ASP came from research on artificial intelligence and computational logic. ASP is a form of declarative programming: an ASP program describes what is counted as a solution to the problem but does not specify an algorithm for solving it. Search is performed by sophisticated software systems called answer set solvers.

Combinatorial search problems often arise in science and technology, and ASP has found applications in diverse areas—in history, linguistics, in bioinformatics, in robotics, in space exploration, in oil and gas industry, and many others. The importance of this programming method was recognized by the Association for the Advancement of Artificial Intelligence in 2016, when AI Magazine published a special issue on answer set programming. The book introduces the reader to the theory and practice of ASP. It describes the input language of the answer set solver CLINGO, which was designed at the University of Potsdam in Germany and is used to solve problems by programmers in many countries. It includes numerous examples of ASP programs and presents the mathematical theory that underlies ASP. There are many exercises with complete solutions.

Proceedings

11th International Conference, CP 2005, Sitges Spain, October 1-5, 2005

Applying Constraint Programming to Scheduling Problems

Constraint Logic Programming

Practical Aspects of Declarative Languages

Essentials of Constraint Programming

Handbook of Constraint Programming Elsevier

This book constitutes the thoroughly refereed and extended post-proceedings of the Joint ERCIM/CoLogNet International Workshop on Constraint Solving and Constraint Logic Programming, CSCLP 2005. The 12 revised full papers presented were carefully reviewed and selected for inclusion in the book. The papers are organized in topical sections on global constraints, search and heuristics, language and implementation issues, and modeling.

Constraint and Integer Programming presents some of the basic ideas of constraint programming and mathematical programming, explores approaches to integration, brings us up to date on heuristic methods, and attempts to discern future directions in this fast-moving field.

This book constitutes the proceedings of the 16th International Conference on Integration of Constraint Programming, Artificial Intelligence, and Operations Research, CPAIOR 2019, held in Thessaloniki, Greece, in June 2019. The 34 full papers presented together with 9 short papers were carefully reviewed and selected from 94 submissions. The conference brings together interested researchers from Constraint Programming (CP), Artificial Intelligence (AI), and Operations Research (OR) to present new techniques or applications and to provide an opportunity for researchers in one area to learn about techniques in the others. A main objective of this conference series is also to give these researchers the opportunity to show how the integration of techniques from different fields can lead to interesting results on large and complex problems.

Selected Research

PPDP.

Distributed Constraint Logic Programming
Principles and Practice of Constraint Programming - CP 2002
Objects for Concurrent Constraint Programming
Constraint Programming and Decision Making

This collection presents the best of current work on all aspects of constraint logic programming languages, from theory through language implementation.

Constraints are everywhere: most computational problems can be described in terms of restrictions imposed on the set of possible solutions, and constraint programming is a problem-solving technique that works by incorporating those restrictions in a programming environment. It draws on methods from combinatorial optimisation and artificial intelligence, and has been successfully applied in a number of fields from scheduling, computational biology, finance, electrical engineering and operations research through to numerical analysis. This textbook for upper-division students provides a thorough and structured account of the main aspects of constraint programming. The author provides many worked examples that illustrate the usefulness and versatility of this approach to programming, as well as many exercises throughout the book that illustrate techniques, test skills and extend the text. Pointers to current research, extensive historical and bibliographic notes, and a comprehensive list of references will also be valuable to professionals in computer science and artificial intelligence.

"Highly recommended to everyone interested in deepening their understanding of Python and practical computer science."

—Daniel Kenney-Jung, MD, University of Minnesota
Key Features Master formal techniques taught in college computer science classes
Connect computer science theory to real-world applications, data, and performance
Prepare for programmer interviews
Recognize the core ideas behind most "new" challenges
Covers Python 3.7
Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.
About The Book Programming problems that seem new or unique are usually rooted in well-known engineering principles. *Classic Computer Science Problems in Python* guides you through time-tested scenarios, exercises, and algorithms that will prepare you for the "new" problems you'll face when you start your next project. In this amazing book, you'll tackle dozens of coding challenges, ranging from simple tasks like binary search algorithms to clustering data using k-means. As you work through examples for web development, machine learning, and more, you'll remember important things you've forgotten and discover classic solutions that will save you hours of time.
What You Will Learn Search algorithms
Common techniques for graphs
Neural networks
Genetic algorithms
Adversarial search
Uses type hints throughout
This Book Is Written For For intermediate Python programmers.
About The Author David Kopec is an assistant professor of Computer Science and Innovation at Champlain College in Burlington, Vermont. He is the author of *Dart for Absolute Beginners* (Apress, 2014), *Classic Computer Science Problems in Swift* (Manning, 2018), and *Classic Computer Science Problems in Java* (Manning, 2020)
Table of Contents Small problems
Search problems
Constraint-satisfaction problems
Graph problems
Genetic algorithms
K-means clustering
Fairly simple neural networks
Adversarial search
Miscellaneous problems

This book constitutes the refereed proceedings of the 8th International Conference on Principles and Practice of Constraint Programming, CP 2002, held in Ithaca, NY, USA in September 2002. The 38 revised full papers and 6 innovative application papers as well as the 14 short papers presented together with 25 abstracts from contributions to the doctoral program were carefully reviewed and selected from 146 submissions. All current issues in constraint processing are addressed, ranging from theoretical and foundational issues to application in various fields.

8th International Conference, CP 2002, Ithaca, NY, USA, September 9-13, 2002, Proceedings

Constraint Programming in Music

Satisfiability Research in the Year 2005

SAT 2005

Classic Computer Science Problems in Python

Answer Set Programming

This book presents the first attempt to combine concurrent logic programming and constraint logic programming. It is divided into three parts. In the first part, a novel computation model, called the multi-Pandora model, which is designed on the basis of the Pandora model, is presented. In the second part, the distributed implementation schemes for Parlog, Pandora, and multi-Pandora are presented. Finally, the author presents the distributed constraint solvers for finite domain constraints, as well as the distributed constraint solvers in the domains of real numbers and Boolean rings which can be incorporated into the schemes presented in the second part to handle the ?ask?- and ?tell?-constraints.

Concurrent constraint programming (ccp) is a recent development in programming language design. Its central contribution is the notion of partial information provided by a shared constraint store. This constraint store serves as a communication medium between concurrent threads of control and as a vehicle for their synchronization. Objects for Concurrent Constraint Programming analyzes the possibility of supporting object-oriented programming in ccp. Starting from established approaches, the book covers various object models and discusses their properties. Small Oz, a sublanguage of the ccp language Oz, is used as a model language for this analysis. This book presents a general-purpose object system for Small Oz and describes its implementation and expressivity for concurrent computation. Objects for Concurrent Constraint Programming is written for programming language researchers with an interest in programming language aspects of concurrency, object-oriented programming, or constraint programming. Programming language implementors will benefit from the rigorous treatment of the efficient implementation of Small Oz. Oz programmers will get a first-hand view of the design decisions that lie behind the Oz object system.

Constraint programming is a powerful paradigm for solving combinatorial search problems that draws on a wide range of techniques from artificial intelligence, computer science, databases, programming languages, and operations research. Constraint programming is currently applied with success to many domains, such as scheduling, planning, vehicle routing, configuration, networks, and bioinformatics. The aim of this handbook is to capture the full breadth and depth of the constraint programming field and to be encyclopedic in its scope and coverage. While there are several excellent books on constraint programming, such books necessarily focus on the main notions and techniques and cannot cover also extensions, applications, and languages. The handbook gives a reasonably complete coverage of all these lines of work, based on constraint programming, so that a reader can have a rather precise idea of the whole field and its potential. Of course each line of work is dealt with in a survey-like style, where some details may be neglected in favor of coverage. However, the extensive bibliography of each chapter will help the interested readers to find suitable sources for the missing details. Each chapter of the handbook is intended to be a self-contained survey of a topic, and is written by one or more authors who are leading researchers in the area. The intended audience of the handbook is researchers, graduate students, higher-year undergraduates and practitioners who wish to learn about the state-of-the-art in constraint programming. No prior knowledge about the field is necessary to be able to read the chapters and gather useful knowledge. Researchers from other fields should find in this handbook an effective way to learn about constraint programming and to possibly use some of the constraint programming concepts and techniques in their work, thus providing a means for a fruitful cross-fertilization among different research areas. The handbook is organized in two parts. The first part covers the basic foundations of constraint programming, including the history, the notion of constraint propagation, basic search methods, global constraints, tractability and computational complexity, and important issues in modeling a problem as a constraint problem. The second part covers constraint languages and solver, several useful extensions to the basic framework (such as interval constraints, structured domains, and distributed CSPs), and successful application areas for constraint programming. - Covers the whole field of constraint programming - Survey-style chapters - Five chapters on applications

The ubiquity of combinatorial optimization problems in our society is illustrated by the novel application areas for optimization technology, which range from supply chain management to sports tournament scheduling. Over the last two decades, constraint programming has emerged as a fundamental methodology to solve a variety of combinatorial problems, and rich constraint programming languages have been developed for expressing and combining constraints and specifying search procedures at a high level of abstraction. Local search approaches to combinatorial optimization are able to isolate optimal or near-optimal solutions within reasonable time constraints. This book introduces a method for solving combinatorial optimization problems that combines constraint programming and local search, using constraints to describe and control local search, and a programming language, COMET, that supports both modeling and search abstractions in the spirit of constraint programming. After an overview of local search including neighborhoods, heuristics, and metaheuristics, the book presents the architecture and modeling and search components of constraint-based local search and describes how constraint-based local search is supported in COMET. The book describes a variety of applications, arranged by meta-heuristics. It presents scheduling applications, along with the background necessary to understand these challenging problems. The book also includes a number of satisfiability problems, illustrating the ability of constraint-based local search approaches to cope with both satisfiability and optimization problems in a uniform fashion.

29th International Symposium, LOPSTR 2019, Porto, Portugal, October 8-10, 2019, Revised Selected Papers

16th International Conference, CPAIOR 2019, Thessaloniki, Greece, June 4-7, 2019, Proceedings

8th International Symposium, PADL 2006, Charleston, SC, USA, January 9-10, 2006, Proceedings

Concurrent Constraint Programming

Constraint Solving and Planning with Picat

Constraint-based Local Search

This book constitutes the thoroughly refereed postproceedings of the 17th International Symposium on Logic-Based Program Synthesis and Transformation, LOPSTR 2007, held in Kongens Lyngby, Denmark, August 23-24, 2007 colocated with SAS 2007. The 13 revised full papers presented together with one invited talk

were carefully selected and revised from 30 submissions during two rounds of reviewing and improvement. The papers are organized in topical sections on program termination, program transformation, constraint solving and analysis as well as software engineering.

First International Conference, CPAIOR 2004, Nice, France, April 20-22, 2004, Proceedings
Constraint Handling in Cohort Intelligence Algorithm
Principles of Constraint Programming