

## Contents

<b>Introduction</b> . . . . .	17
Frédéric BENHAMOU, Narendra JUSSIEN, Barry O’SULLIVAN	
<b>Part I. The Past, Present and Future of Constraint Programming</b> . . .	23
Frédéric BENHAMOU, Narendra JUSSIEN , Barry O’SULLIVAN	
<b>Chapter 1. Constraint Programming as Declarative Algorithmics</b> . . .	25
Pascal VAN HENTENRYCK	
1.1. The CHIP project . . . . .	26
1.2. The Numerica project . . . . .	32
1.3. The OPL project . . . . .	34
1.4. The Comet project . . . . .	35
1.5. The future of constraint programming . . . . .	38
<b>Chapter 2. Constraint Programming Tools</b> . . . . .	41
Laurent MICHEL, Christian SCHULTE, Pascal VAN HENTENRYCK	
2.1. Introduction . . . . .	41
2.2. Invited talks . . . . .	43
2.2.1. The development of an industrial CP tool (Jean-François Puget) . . . . .	43
2.2.1.1. Design goals . . . . .	44
2.2.1.2. The future . . . . .	45
2.2.2. System design: taking informed decisions (Christian Schulte)	45
2.2.2.1. Search . . . . .	46
2.2.2.2. Constraint propagation . . . . .	46
2.2.2.3. Variables . . . . .	47
2.3. System presentations . . . . .	48
2.3.1. ECL <sup>i</sup> PS <sup>e</sup> . . . . .	48
2.3.2. SICStus FD . . . . .	48

6 Trends in Constraint Programming

2.3.3. G12 . . . . .	49
2.3.4. DiSolver . . . . .	49
2.3.5. MINION . . . . .	50
2.3.6. Choco . . . . .	50
2.3.7. Gecode . . . . .	51
2.3.8. Comet . . . . .	52
2.3.9. JaCoP . . . . .	53
2.3.10. Borderwijk . . . . .	54
2.4. Panels . . . . .	54
2.5. Conclusion . . . . .	56
2.6. References . . . . .	57
<b>Chapter 3. The Next 10 Years of Constraint Programming</b> . . . . .	<b>59</b>
Lucas BORDEAUX, Barry O’SULLIVAN, Pascal VAN HENTENRYCK	
3.1. Pedro Barahona . . . . .	61
3.2. Christian Bessiere . . . . .	63
3.3. Peter Jeavons . . . . .	64
3.4. Pedro Meseguer . . . . .	66
3.5. Gilles Pesant . . . . .	68
3.6. Francesca Rossi . . . . .	70
3.7. Thomas Schiex . . . . .	72
3.8. Christian Schulte . . . . .	74
3.9. Meinolf Sellmann . . . . .	75
3.10. Mark Wallace . . . . .	77
3.11. Toby Walsh . . . . .	79
3.12. Roland Yap . . . . .	80
3.13. References . . . . .	81
<b>Chapter 4. Constraint Propagation and Implementation</b> . . . . .	<b>83</b>
Marc VAN DONGEN, Christophe LECOUTRE	
4.1. Filtering algorithms for precedence and dependency constraints (by Roman Barták and Ondřej Čepek) . . . . .	84
4.1.1. Problem description and related works . . . . .	84
4.1.2. Filtering rules for precedence and dependency constraints . . . . .	85
4.1.3. Summary . . . . .	87
4.2. A study of residual supports in arc consistency (by Christophe Lecoutre and Fred Hemery) . . . . .	87
4.3. Maintaining singleton arc consistency (by Christophe Lecoutre and Patrick Prosser) . . . . .	89
4.3.1. Mixed consistency . . . . .	90
4.3.2. Checking existential-SAC . . . . .	91
4.3.3. Conclusion . . . . .	92

4.4. Probabilistic singleton arc consistency (by Deepak Mehta and Marc van Dongen) . . . . .	93
4.5. Simplification and extension of the SPREAD constraint (by Pierre Schaus, Yves Deville, Pierre Dupont and Jean-Charles Régin)	95
4.5.1. Filtering of $\pi$ . . . . .	96
4.5.2. Filtering of $X$ . . . . .	97
4.5.3. Conclusion . . . . .	99
4.6. A new filtering algorithm for the graph isomorphism problem (by Sébastien Sorlin and Christine Solnon) . . . . .	99
4.6.1. A global constraint for the graph isomorphism problem . .	99
4.6.2. ILL-consistency and ILL-filtering . . . . .	100
4.6.3. Experimental results . . . . .	102
4.7. References . . . . .	103
<b>Chapter 5. On the First SAT/CP Integration Workshop</b> . . . . .	105
Youssef HAMADI, Lucas BORDEAUX	
5.1. The technical program . . . . .	106
5.1.1. The invited talk . . . . .	106
5.1.2. Contributions related to SMT and solver integration . . . .	106
5.1.3. Contributions related to the use of SAT techniques to im- prove CSP/CP solvers . . . . .	107
5.1.4. Other contributions . . . . .	108
5.2. The panel session . . . . .	109
5.2.1. Are SAT and CP different or similar? . . . . .	109
5.2.2. Why has SAT succeeded in reducing the tuning issue? . . .	111
5.2.3. How long can the current generation of SAT solvers evolve? 5.2.4. Were performance issues correctly addressed by CP? . . . .	113
5.2.5. Was CP too ambitious? . . . . .	118
5.2.6. Do we still need CP? . . . . .	119
5.3. Summary, future directions and conclusion . . . . .	121
5.4. References . . . . .	122
<b>Chapter 6. Constraint-Based Methods for Bioinformatics</b> . . . . .	125
Alessandro DAL PALÙ, Agostino DOVIER, François FAGES, Sebastian WILL	
6.1. On using temporal logic with constraints to express biological properties of cell processes (by François Fages) . . . . .	126
6.2. Modeling biological systems in stochastic concurrent constraint programming (by Luca Bortolussi and Alberto Policriti) . . . . .	129
6.3. Chemera: constraints in protein structural problems (by Pedro Barahona and Ludwig Krippahl) . . . . .	132

8 Trends in Constraint Programming

6.4. Exploiting model checking in constraint-based approaches to the protein folding problem (by Elisabetta De Maria, Agostino Dovier, Angelo Montanari and Carla Piazza) . . . . .	134
6.5. Global constraints for discrete lattices (by Alessandro Dal Palù, Agostino Dovier and Enrico Pontelli) . . . . .	136
6.6. Counting protein structures by DFS with dynamic decomposition (by Sebastian Will and Martin Mann) . . . . .	138
6.7. Suffix array and weighted CSPs (by Matthias Zytnicki, Christine Gaspin and Thomas Schiex) . . . . .	141
6.8. Supertree construction with constraint programming: recent progress and new challenges (by Patrick Prosser) . . . . .	143
6.9. References . . . . .	145

**Part II. Constraint Modeling and Reformulation** . . . . . 147  
Ian MIGUEL, Steven PRESTWICH

**Chapter 7. Improved Models for Graceful Graphs** . . . . . 151  
Jean-François PUGET, Barbara SMITH

7.1. Introduction . . . . .	151
7.2. A direct model . . . . .	152
7.3. The edge-label model . . . . .	154
7.4. A combined model . . . . .	156
7.5. Experimental results . . . . .	157
7.6. Discussion . . . . .	160
7.7. References . . . . .	161

**Chapter 8. The Automatic Generation of Redundant Representations and Channeling Constraints** . . . . . 163  
Bernadette MARTÍNEZ-HERNÁNDEZ, Alan M. FRISCH

8.1. Introduction . . . . .	163
8.2. Representations . . . . .	167
8.3. Alternative representations and channels . . . . .	168
8.3.1. Alternative representations . . . . .	168
8.3.2. Constraint-wise quasi-representations and channeling constraints . . . . .	169
8.4. Refinement . . . . .	174
8.5. Systematic generation of channeling constraints . . . . .	177
8.6. Producing the best alternative for channeling . . . . .	179
8.7. Conclusions and future work . . . . .	180
8.8. References . . . . .	180

**Part III. Symmetry in Constraint Satisfaction Problems** . . . . . 183  
Alastair DONALDSON, Peter GREGORY, Karen PETRIE

<b>Chapter 9. GAPLex: Generalized Static Symmetry Breaking</b> . . . . .	187
Chris JEFFERSON, Tom KELSEY, Steve LINTON, Karen PETRIE	
9.1. Background and introduction . . . . .	188
9.1.1. Group theory for CSPs . . . . .	190
9.1.2. Using GAP to break CSP symmetries . . . . .	191
9.2. GAPLex . . . . .	192
9.2.1. Motivation and rationale . . . . .	192
9.2.2. Motivating example . . . . .	193
9.2.3. The GAPLex algorithms . . . . .	193
9.3. Empirical evaluation . . . . .	196
9.3.1. Combining GAPLex with incomplete static SB methods . . . . .	197
9.3.2. Combining GAPLex with Puget’s all-different constraints . . . . .	198
9.4. Conclusions and future work . . . . .	199
9.5. References . . . . .	199
<b>Chapter 10. Symmetry Breaking in Subgraph Pattern Matching</b> . . . . .	203
Stéphane ZAMPELLI, Yves DEVILLE, Pierre DUPONT	
10.1. Background and definitions . . . . .	205
10.2. Variable symmetries . . . . .	207
10.2.1. Detection . . . . .	207
10.2.2. Breaking . . . . .	207
10.3. Value symmetries . . . . .	208
10.3.1. Detection . . . . .	208
10.3.2. Breaking . . . . .	208
10.4. Experimental results . . . . .	209
10.5. Local value symmetries . . . . .	211
10.5.1. Dynamic target graph . . . . .	212
10.5.1.1. Detection . . . . .	212
10.5.1.2. Breaking . . . . .	213
10.5.2. Partial dynamic graphs . . . . .	214
10.5.2.1. Detection . . . . .	214
10.5.2.2. Breaking . . . . .	215
10.6. Conclusion . . . . .	215
10.7. References . . . . .	216
<b>Part IV. Interval Analysis, Constraint Propagation and Applications</b> . . . . .	219
Christophe JERMANN, Yahia LEBBAH, Djamila SAM-HAROUD	
<b>Chapter 11. Modeling and Solving of a Radio Antenna Deployment Support Application</b> . . . . .	223
Michael HEUSCH	
11.1. Two simple models for the application . . . . .	224

11.1.1. A first finite domain model . . . . .	224
11.1.2. Shifting the model to mixed domains . . . . .	225
11.1.3. Description of the search algorithm . . . . .	225
11.1.4. Analysis of the performance on progressive deployment problems . . . . .	226
11.2. Introducing the <i>distn</i> constraint . . . . .	227
11.3. Modeling the application with the <i>distn</i> constraint . . . . .	228
11.3.1. Revised model of the application . . . . .	228
11.3.2. Numerical results when solving the LocRLFAP with <i>distn</i>	230
11.3.3. Qualitative analysis of the results . . . . .	231
11.4. Conclusion . . . . .	231
11.5. References . . . . .	232
<b>Chapter 12. Guaranteed Numerical Injectivity Test via Interval Analysis</b>	233
Sébastien LAGRANGE, Nicolas DELANOUE, Luc JAULIN	
12.1. Interval analysis . . . . .	235
12.2. Injectivity . . . . .	236
12.2.1. Partial injectivity . . . . .	236
12.2.2. Partial injectivity condition . . . . .	238
12.3. ITVIA algorithm . . . . .	241
12.4. Examples . . . . .	242
12.4.1. Spiral function . . . . .	243
12.4.2. Ribbon function . . . . .	243
12.5. Conclusion . . . . .	244
12.6. References . . . . .	244
<b>Chapter 13. An Interval-based Approximation Method for Discrete Changes in Hybrid cc</b> . . . . .	245
Daisuke ISHII, Kazunori UEDA, Hiroshi HOSOBÉ	
13.1. An overview of Hybrid cc . . . . .	246
13.1.1. The Hybrid cc language . . . . .	246
13.1.2. Implementation of Hybrid cc . . . . .	247
13.2. The objective of the chapter . . . . .	248
13.3. Background of interval arithmetic . . . . .	248
13.3.1. Basic notions of interval arithmetic . . . . .	249
13.3.2. ODE solving based on interval arithmetic . . . . .	249
13.4. The proposed method . . . . .	249
13.4.1. Assumptions on the proposed method . . . . .	249
13.4.2. TRACE algorithm . . . . .	250
13.4.3. PRUNEANDMERGE algorithm . . . . .	251
13.5. Experimental results . . . . .	252
13.6. Related work . . . . .	253
13.7. Conclusion . . . . .	254

13.8. References . . . . .	254
<b>Part V. Local Search Techniques in Constraint Satisfaction</b> . . . . .	257
Andrea ROLI, Yehuda NAVEH	
<b>Chapter 14. Combining Adaptive Noise and Look-Ahead in Local Search for SAT</b> . . . . .	261
Chu Min LI, Wanxia WEI, Harry ZHANG	
14.1. Implementation of the adaptive noise mechanism in $G^2WSAT$	262
14.2. Look-Ahead for promising decreasing variables . . . . .	262
14.2.1. Promising score of a variable . . . . .	262
14.2.2. Integrating limited look-ahead in $adaptG^2WSAT$ . . . . .	263
14.3. Evaluation . . . . .	264
14.4. Conclusion . . . . .	266
14.5. References . . . . .	266
<b>Chapter 15. Finding Large Cliques using SAT Local Search</b> . . . . .	269
Steven PRESTWICH	
15.1. SAT-encoding the clique problem . . . . .	270
15.2. Notes on the bitwise at-most-one encoding . . . . .	271
15.3. Experiments . . . . .	272
15.4. Conclusion . . . . .	273
15.5. References . . . . .	274
<b>Chapter 16. Multi-Point Constructive Search for Constraint Satisfaction: An Overview</b> . . . . .	275
Ivan HECKMAN, J. Christopher BECK	
16.1. Background . . . . .	276
16.2. Empirical study . . . . .	277
16.3. Conclusion . . . . .	279
16.4. References . . . . .	280
<b>Chapter 17. Boosting SLS Using Resolution</b> . . . . .	283
ANBULAGAN, Duc Nghia PHAM, John SLANEY, Abdul SATTAR	
17.1. SLS solvers . . . . .	284
17.2. Preprocessors . . . . .	285
17.3. Empirical evaluation . . . . .	286
17.3.1. Clause weighting versus random walk . . . . .	286
17.3.2. Matching preprocessors to solver-problem pairs . . . . .	287
17.3.3. Multiple preprocessing and preprocessor ordering . . . . .	287
17.4. Conclusion . . . . .	288
17.5. References . . . . .	288

<b>Chapter 18. Growing COMET</b> . . . . .	291
Pascal VAN HENTENRYCK, Laurent MICHEL	
18.1. Constraint-based local search . . . . .	291
18.2. COMET . . . . .	292
18.3. Modeling . . . . .	293
18.4. Search . . . . .	295
18.5. References . . . . .	296
<b>Part VI. Preferences and Soft Constraints</b> . . . . .	299
Thomas SCHIEX	
<b>Chapter 19. The Logic Behind Weighted CSP</b> . . . . .	303
Carlos ANSÓTEGUI, María L. BONET, Jordi LEVY, Felip MANYÀ	
19.1. Preliminaries . . . . .	304
19.2. The inference rule – soundness and completeness . . . . .	307
19.3. Global consistency in WCSP . . . . .	310
19.4. Local consistency in WCSP . . . . .	311
19.5. Conclusions . . . . .	314
19.6. References . . . . .	316
<b>Chapter 20. Dynamic Heuristics for Branch and Bound on Tree-Decomposition of Weighted CSPs</b> . . . . .	317
Philippe JÉGOU, Samba Ndojh NDIAYE, Cyril TERRIOUX	
20.1. Introduction . . . . .	317
20.2. Preliminaries . . . . .	319
20.3. Dynamic orders in $O(\exp(w + 1))$ . . . . .	322
20.4. Bounded extensions of dynamic orders . . . . .	324
20.5. Heuristics . . . . .	325
20.5.1. Cluster orders . . . . .	325
20.5.2. Variable orders . . . . .	327
20.5.3. Heuristics for grouping variables in <i>Classes 4</i> and <i>5</i> . . . . .	327
20.6. Experimental study . . . . .	328
20.7. Discussion and conclusion . . . . .	330
20.8. References . . . . .	331
<b>Part VII. Constraints in Software Testing, Verification and Analysis</b> . . . . .	333
Benjamin BLANC, Arnaud GOTLIEB, Claude MICHEL	
<b>Chapter 21. Extending a CP Solver with Congruences as Domains for Program Verification</b> . . . . .	337
Michel LECONTE, Bruno BERSTEL	
21.1. Related work . . . . .	339

21.2. Congruences as domains . . . . .	339
21.3. Propagation of congruences as domains . . . . .	340
21.4. Cooperation of congruences and intervals . . . . .	342
21.5. Conclusion . . . . .	342
21.6. References . . . . .	343
<b>Chapter 22. Generating Random Values Using Binary Decision Diagrams and Convex Polyhedra . . . . .</b>	<b>345</b>
Erwan JAHIER, Pascal RAYMOND	
22.1. BDD and convex polyhedra . . . . .	346
22.2. The resolution algorithm . . . . .	346
22.3. Choosing solutions . . . . .	348
22.4. Available tools . . . . .	349
22.5. Related work . . . . .	350
22.6. Conclusion . . . . .	351
22.7. References . . . . .	351
<b>Chapter 23. A Symbolic Model for Hash-Collision Attacks . . . . .</b>	<b>353</b>
Yannick CHEVALIER, Mounira KOURJIEH	
23.1. Terms and subterms . . . . .	354
23.2. Analysis of reachability properties of cryptographic protocols . . . . .	356
23.3. Model of a collision-aware intruder . . . . .	357
23.3.1. Intruder on words . . . . .	357
23.3.2. Intruder on words with free function symbols . . . . .	358
23.3.3. Hash-colliding intruder . . . . .	358
23.4. Conclusion . . . . .	359
23.5. References . . . . .	359
<b>Chapter 24. Strategy for Flaw Detection Based on a Service-driven Model for Group Protocols . . . . .</b>	<b>361</b>
Najah CHRIDI, Laurent VIGNERON	
24.1. Protocol modeling and attack search . . . . .	362
24.1.1. Input of the method . . . . .	362
24.1.2. Searching for attacks in group protocols . . . . .	363
24.1.3. Intruder knowledge management . . . . .	365
24.1.4. Constraint management . . . . .	366
24.2. Verification results . . . . .	366
24.3. Summary and future work . . . . .	367
24.4. References . . . . .	368
<b>Part VIII. Constraint Programming for Graphical Applications . . . . .</b>	<b>369</b>
Marc CHRISTIE, Hiroshi HOSOBÉ and Kim MARRIOTT	

<b>Chapter 25. Trends and Issues in using Constraint Programming for Graphical Applications</b> . . . . .	371
Marc CHRISTIE, Hiroshi HOSOBÉ and Kim MARRIOTT	
25.1. More powerful constraint-solving techniques . . . . .	373
25.1.1. Mixture of discrete and continuous constraints . . . . .	373
25.1.2. Mixture of linear, polynomial, geometric and non-linear constraints . . . . .	373
25.1.3. Managing user interaction . . . . .	374
25.1.4. Managing preferences . . . . .	374
25.1.5. Generic techniques . . . . .	375
25.2. Better modeling and understanding of constraint systems by the end-user . . . . .	376
25.2.1. Model specification . . . . .	376
25.2.2. Extensibility . . . . .	377
25.2.3. Constraint representation . . . . .	377
25.2.4. Understanding constraint interaction during solving . . . . .	377
25.3. Bridging the gap between the solver and the application semantics . . . . .	378
25.3.1. High-level modeling . . . . .	379
25.3.2. Support for interaction . . . . .	379
25.4. Conclusion . . . . .	379
25.5. References . . . . .	380
<b>Chapter 26. A Constraint Satisfaction Framework for Visual Problem Solving</b> . . . . .	383
Bonny BANERJEE, Balakrishnan CHANDRASEKARAN	
26.1. The framework . . . . .	384
26.1.1. A language for expressing visual problems . . . . .	384
26.1.1.1. The specification language . . . . .	384
26.1.1.2. The modeling language . . . . .	387
26.1.2. A visual problem solver . . . . .	388
26.2. Applications . . . . .	390
26.3. Conclusion . . . . .	393
26.4. References . . . . .	393
<b>Chapter 27. Computer Graphics and Constraint Solving: An Application to Virtual Camera Control</b> . . . . .	395
Jean-Marie NORMAND	
27.1. Overview . . . . .	397
27.2. A semantic space partitioning approach . . . . .	398
27.2.1. Projection property . . . . .	398
27.2.2. Orientation property . . . . .	399
27.2.3. Occlusion property . . . . .	399

27.3. Numerical solving stage . . . . .	400
27.4. Exploitation of semantic volumes . . . . .	401
27.4.1. Making requests on the volumes . . . . .	401
27.4.2. Making requests on the scene . . . . .	401
27.5. Results . . . . .	401
27.6. Discussion . . . . .	403
27.7. References . . . . .	404
<b>Index</b> . . . . .	<b>405</b>