Krzysztof R. Apt
François Fages
Francesca Rossi
Péter Szeredi
József Váncza (Eds.)

LNAI 3010

Recent Advances in Constraints

Joint ERCIM/CoLogNET International Workshop on Constraint Solving and Constraint Logic Programming, CSCLP 2003
Budapest, Hungary, June/July 2003, Selected Papers

Springer

Krzysztof R. Apt François Fages
Francesca Rossi Péter Szeredi
József Váncza (Eds.)

Recent Advances in Constraints

Joint ERCIM/CoLogNET International Workshop
on Constraint Solving and Constraint Logic Programming, CSCLP 2003
Budapest, Hungary, June 30 - July 2, 2003
Selected Papers

Springer

Series Editors

Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editors

Krzysztof R. Apt
National University of Singapore, School of Computing
3 Science Drive 2, Republic of Singapore 117543
E-mail: apt@comp.nus.edu.sg

François Fages
INRIA Rocquencourt - Projet CONTRAINTES
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex, France
E-mail: Francois.Fages@inria.fr

Francesca Rossi
University of Padua, Department of Pure and Applied Mathematics
Via G.B. Belzoni 7, 35131 Padua, Italy
E-mail: frossi@math.unipd.it

Péter Szeredi
Budapest University of Technology and Economics
Department of Computer Science and Information Theory
Magyar tudosok krt. 2, 1117 Budapest, Hungary
E-mail: szeredi@cs.bme.hu

József Váncza
Hungarian Academy of Sciences, Computer and Automation Research Institute
Kende u. 13-17, 1111 Budapest, Hungary
E-mail: vancza@sztaki.hu

Library of Congress Control Number: 2004103751

CR Subject Classification (1998): I.2.3-4, D.1, D.3.2-3, F.3.2, F.4.1, I.2.8, F.2.2

ISSN 0302-9743
ISBN 3-540-21834-3 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN: 10997857 06/3142 5 4 3 2 1 0

Lecture Notes in Artificial Intelligence 3010
Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Springer
Berlin
Heidelberg
New York
Hong Kong
London
Milan
Paris
Tokyo

Preface

Constraint programming is the fruit of several decades of research carried out in mathematical logic, automated deduction, operations research and artificial intelligence. The tools and programming languages arising from this research field have enjoyed real success in the industrial world as they contribute to solving hard combinatorial problems in diverse domains such as production planning, communication networks, robotics and bioinformatics.

This volume contains the extended and reviewed versions of a selection of papers presented at the Joint ERCIM/CoLogNET International Workshop on Constraint Solving and Constraint Logic Programming (CSCLP 2003), which was held from June 30 to July 2, 2003. The venue chosen for the seventh edition of this annual workshop was the Computer and Automation Research Institute of the Hungarian Academy of Sciences (MTA SZTAKI) in Budapest, Hungary. This institute is one of the 20 members of the Working Group on Constraints of the European Research Consortium for Informatics and Mathematics (ERCIM). For many participants this workshop provided the first opportunity to visit their ERCIM partner in Budapest.

CoLogNET is the European-funded network of excellence dedicated to supporting and enhancing cooperation and research on all areas of computational logic, and continues the work done previously by the Compulog Net. In particular, the aim of the logic and constraint logic programming area of CoLogNET is to foster and support all research activities related to logic programming and constraint logic programming.

The editors would like to take the opportunity and thank all the authors who submitted papers to this volume, as well as the reviewers for their helpful work. We hope this volume is useful for anyone interested in the recent advances and new trends in constraint programming, constraint solving, problem modelling and applications.

January 2004 K.R. Apt, F. Fages, F. Rossi, P. Szeredi and J. Váncza

Organization

CSCLP 2003 was organized by the ERCIM Working Group on Constraints, the CoLogNET area on Logic and Constraint Logic Programming, and the MTA SZTAKI in Budapest.

Organizing and Program Committee

Krzysztof Apt	University of Singapore, Singapore
François Fages	INRIA Rocquencourt, France
Francesca Rossi	University of Padua, Italy
Péter Szeredi	BUTE, Budapest, Hungary
József Váncza	MTA SZTAKI, Budapest, Hungary

Referees

K.R. Apt
R. Backofen
R. Barták
F. Benhamou
T. Benkő
S. Bistarelli
L. Bordeaux
S. Brand
M. Carlsson
M. Ceberio
N. Chabrier-Rivier
D. Diaz
F. Fages
B. Faltings
R. Gennari
W.J. van Hoeve
T. Kis
A. Lallouet
A. Márkus
E. Monfroy
B. O'Sullivan
G. Ringwelski
F. Rossi
H. Rudová
Zs. Ruttkay
H. Simonis
S. Soliman
P. Szeredi
J. Váncza
T. Walsh
A. Wolf

Sponsoring Institutions

ERCIM Working Group on Constraints
European Network of Excellence CoLogNET
Ministry of Education Hungary
MTA SZTAKI

Lecture Notes in Artificial Intelligence (LNAI)

Vol. 3010: K.R. Apt, F. Fages, F. Rossi, P. Szeredi, J. Váncza (Eds.), Recent Advances in Constraints. VIII, 285 pages. 2004.

Vol. 2980: A. Blackwell, K. Marriott, A. Shimojima (Eds.), Diagrammatic Representation and Inference. XV, 448 pages. 2004.

Vol. 2977: G. Di Marzo Serugendo, A. Karageorgos, O.F. Rana, F. Zambonelli (Eds.), Engineering Self-Organising Systems. X, 299 pages. 2004.

Vol. 2972: R. Monroy, G. Arroyo-Figueroa, L.E. Sucar, H. Sossa (Eds.), MICAI 2004: Advances in Artificial Intelligence. XVII, 923 pages. 2004.

Vol. 2961: P. Eklund (Ed.), Concept Lattices. IX, 411 pages. 2004.

Vol. 2953: K. Konrad, Model Generation for Natural Language Interpretation and Analysis. XIII, 166 pages. 2004.

Vol. 2934: G. Lindemann, D. Moldt, M. Paolucci (Eds.), Regulated Agent-Based Social Systems. X, 301 pages. 2004.

Vol. 2930: F. Winkler (Ed.), Automated Deduction in Geometry. VII, 231 pages. 2004.

Vol. 2926: L. van Elst, V. Dignum, A. Abecker (Eds.), Agent-Mediated Knowledge Management. XI, 428 pages. 2004.

Vol. 2923: V. Lifschitz, I. Niemelä (Eds.), Logic Programming and Nonmonotonic Reasoning. IX, 365 pages. 2004.

Vol. 2915: A. Camurri, G. Volpe (Eds.), Gesture-Based Communication in Human-Computer Interaction. XIII, 558 pages. 2004.

Vol. 2913: T.M. Pinkston, V.K. Prasanna (Eds.), High Performance Computing - HiPC 2003. XX, 512 pages. 2003.

Vol. 2903: T.D. Gedeon, L.C.C. Fung (Eds.), AI 2003: Advances in Artificial Intelligence. XVI, 1075 pages. 2003.

Vol. 2902: F.M. Pires, S.P. Abreu (Eds.), Progress in Artificial Intelligence. XV, 504 pages. 2003.

Vol. 2892: F. Dau, The Logic System of Concept Graphs with Negation. XI, 213 pages. 2003.

Vol. 2891: J. Lee, M. Barley (Eds.), Intelligent Agents and Multi-Agent Systems. X, 215 pages. 2003.

Vol. 2882: D. Veit, Matchmaking in Electronic Markets. XV, 180 pages. 2003.

Vol. 2871: N. Zhong, Z.W. Raś, S. Tsumoto, E. Suzuki (Eds.), Foundations of Intelligent Systems. XV, 697 pages. 2003.

Vol. 2854: J. Hoffmann, Utilizing Problem Structure in Planing. XIII, 251 pages. 2003.

Vol. 2843: G. Grieser, Y. Tanaka, A. Yamamoto (Eds.), Discovery Science. XII, 504 pages. 2003.

Vol. 2842: R. Gavaldá, K.P. Jantke, E. Takimoto (Eds.), Algorithmic Learning Theory. XI, 313 pages. 2003.

Vol. 2838: N. Lavrač, D. Gamberger, L. Todorovski, H. Blockeel (Eds.), Knowledge Discovery in Databases: PKDD 2003. XVI, 508 pages. 2003.

Vol. 2837: N. Lavrač, D. Gamberger, L. Todorovski, H. Blockeel (Eds.), Machine Learning: ECML 2003. XVI, 504 pages. 2003.

Vol. 2835: T. Horváth, A. Yamamoto (Eds.), Inductive Logic Programming. X, 401 pages. 2003.

Vol. 2821: A. Günter, R. Kruse, B. Neumann (Eds.), KI 2003: Advances in Artificial Intelligence. XII, 662 pages. 2003.

Vol. 2807: V. Matoušek, P. Mautner (Eds.), Text, Speech and Dialogue. XIII, 426 pages. 2003.

Vol. 2801: W. Banzhaf, J. Ziegler, T. Christaller, P. Dittrich, J.T. Kim (Eds.), Advances in Artificial Life. XVI, 905 pages. 2003.

Vol. 2797: O.R. Zaïane, S.J. Simoff, C. Djeraba (Eds.), Mining Multimedia and Complex Data. XII, 281 pages. 2003.

Vol. 2792: T. Rist, R. Aylett, D. Ballin, J. Rickel (Eds.), Intelligent Virtual Agents. XV, 364 pages. 2003.

Vol. 2782: M. Klusch, A. Omicini, S. Ossowski, H. Laamanen (Eds.), Cooperative Information Agents VII. XI, 345 pages. 2003.

Vol. 2780: M. Dojat, E. Keravnou, P. Barahona (Eds.), Artificial Intelligence in Medicine. XIII, 388 pages. 2003.

Vol. 2777: B. Schölkopf, M.K. Warmuth (Eds.), Learning Theory and Kernel Machines. XIV, 746 pages. 2003.

Vol. 2752: G.A. Kaminka, P.U. Lima, R. Rojas (Eds.), RoboCup 2002: Robot Soccer World Cup VI. XVI, 498 pages. 2003.

Vol. 2741: F. Baader (Ed.), Automated Deduction – CADE-19. XII, 503 pages. 2003.

Vol. 2705: S. Renals, G. Grefenstette (Eds.), Text- and Speech-Triggered Information Access. VII, 197 pages. 2003.

Vol. 2703: O.R. Zaïane, J. Srivastava, M. Spiliopoulou, B. Masand (Eds.), WEBKDD 2002 - MiningWeb Data for Discovering Usage Patterns and Profiles. IX, 181 pages. 2003.

Vol. 2700: M.T. Pazienza (Ed.), Extraction in the Web Era. XIII, 163 pages. 2003.

Vol. 2699: M.G. Hinchey, J.L. Rash, W.F. Truszkowski, C.A. Rouff, D.F. Gordon-Spears (Eds.), Formal Approaches to Agent-Based Systems. IX, 297 pages. 2002.

Vol. 2691: V. Mařík, J.P. Müller, M. Pechoucek (Eds.), Multi-Agent Systems and Applications III. XIV, 660 pages. 2003.

Vol. 2684: M.V. Butz, O. Sigaud, P. Gérard (Eds.), Anticipatory Behavior in Adaptive Learning Systems. X, 303 pages. 2003.

Vol. 2671: Y. Xiang, B. Chaib-draa (Eds.), Advances in Artificial Intelligence. XIV, 642 pages. 2003.

Vol. 2663: E. Menasalvas, J. Segovia, P.S. Szczepaniak (Eds.), Advances in Web Intelligence. XII, 350 pages. 2003.

Vol. 2661: P.L. Lanzi, W. Stolzmann, S.W. Wilson (Eds.), Learning Classifier Systems. VII, 231 pages. 2003.

Vol. 2654: U. Schmid, Inductive Synthesis of Functional Programs. XXII, 398 pages. 2003.

Vol. 2650: M.-P. Huget (Ed.), Communications in Multiagent Systems. VIII, 323 pages. 2003.

Vol. 2645: M.A. Wimmer (Ed.), Knowledge Management in Electronic Government. XI, 320 pages. 2003.

Vol. 2639: G. Wang, Q. Liu, Y. Yao, A. Skowron (Eds.), Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing. XVII, 741 pages. 2003.

Vol. 2637: K.-Y. Whang, J. Jeon, K. Shim, J. Srivastava, Advances in Knowledge Discovery and Data Mining. XVIII, 610 pages. 2003.

Vol. 2636: E. Alonso, D. Kudenko, D. Kazakov (Eds.), Adaptive Agents and Multi-Agent Systems. XIV, 323 pages. 2003.

Vol. 2627: B. O'Sullivan (Ed.), Recent Advances in Constraints. X, 201 pages. 2003.

Vol. 2600: S. Mendelson, A.J. Smola (Eds.), Advanced Lectures on Machine Learning. IX, 259 pages. 2003.

Vol. 2592: R. Kowalczyk, J.P. Müller, H. Tianfield, R. Unland (Eds.), Agent Technologies, Infrastructures, Tools, and Applications for E-Services. XVII, 371 pages. 2003.

Vol. 2586: M. Klusch, S. Bergamaschi, P. Edwards, P. Petta (Eds.), Intelligent Information Agents. VI, 275 pages. 2003.

Vol. 2583: S. Matwin, C. Sammut (Eds.), Inductive Logic Programming. X, 351 pages. 2003.

Vol. 2581: J.S. Sichman, F. Bousquet, P. Davidsson (Eds.), Multi-Agent-Based Simulation. X, 195 pages. 2003.

Vol. 2577: P. Petta, R. Tolksdorf, F. Zambonelli (Eds.), Engineering Societies in the Agents World III. X, 285 pages. 2003.

Vol. 2569: D. Karagiannis, U. Reimer (Eds.), Practical Aspects of Knowledge Management. XIII, 648 pages. 2002.

Vol. 2560: S. Goronzy, Robust Adaptation to Non-Native Accents in Automatic Speech Recognition. XI, 144 pages. 2002.

Vol. 2557: B. McKay, J. Slaney (Eds.), AI 2002: Advances in Artificial Intelligence. XV, 730 pages. 2002.

Vol. 2554: M. Beetz, Plan-Based Control of Robotic Agents. XI, 191 pages. 2002.

Vol. 2543: O. Bartenstein, U. Geske, M. Hannebauer, O. Yoshie (Eds.), Web Knowledge Management and Decision Support. X, 307 pages. 2003.

Vol. 2541: T. Barkowsky, Mental Representation and Processing of Geographic Knowledge. X, 174 pages. 2002.

Vol. 2533: N. Cesa-Bianchi, M. Numao, R. Reischuk (Eds.), Algorithmic Learning Theory. XI, 415 pages. 2002.

Vol. 2531: J. Padget, O. Shehory, D. Parkes, N.M. Sadeh, W.E. Walsh (Eds.), Agent-Mediated Electronic Commerce IV. Designing Mechanisms and Systems. XVII, 341 pages. 2002.

Vol. 2527: F.J. Garijo, J.-C. Riquelme, M. Toro (Eds.), Advances in Artificial Intelligence - IBERAMIA 2002. XVIII, 955 pages. 2002.

Vol. 2522: T. Andreasen, A. Motro, H. Christiansen, H.L. Larsen (Eds.), Flexible Query Answering Systems. X, 383 pages. 2002.

Vol. 2514: M. Baaz, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning. XIII, 465 pages. 2002.

Vol. 2507: G. Bittencourt, G.L. Ramalho (Eds.), Advances in Artificial Intelligence. XIII, 417 pages. 2002.

Vol. 2504: M.T. Escrig, F. Toledo, E. Golobardes (Eds.), Topics in Artificial Intelligence. XI, 427 pages. 2002.

Vol. 2499: S.D. Richardson (Ed.), Machine Translation: From Research to Real Users. XXI, 254 pages. 2002.

Vol. 2484: P. Adriaans, H. Fernau, M. van Zaanen (Eds.), Grammatical Inference: Algorithms and Applications. IX, 315 pages. 2002.

Vol. 2479: M. Jarke, J. Koehler, G. Lakemeyer (Eds.), KI 2002: Advances in Artificial Intelligence. XIII, 327 pages. 2002.

Vol. 2475: J.J. Alpigini, J.F. Peters, A. Skowron, N. Zhong (Eds.), Rough Sets and Current Trends in Computing. XV, 640 pages. 2002.

Vol. 2473: A. Gómez-Pérez, V.R. Benjamins (Eds.), Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web. XI, 402 pages. 2002.

Vol. 2466: M. Beetz, J. Hertzberg, M. Ghallab, M.E. Pollack (Eds.), Advances in Plan-Based Control of Robotic Agents. VIII, 291 pages. 2002.

Vol. 2464: M. O'Neill, R.F.E. Sutcliffe, C. Ryan, M. Eaton, N.J.L. Griffith (Eds.), Artificial Intelligence and Cognitive Science. XI, 247 pages. 2002.

Vol. 2448: P. Sojka, I. Kopecek, K. Pala (Eds.), Text, Speech and Dialogue. XII, 481 pages. 2002.

Vol. 2447: D.J. Hand, N.M. Adams, R.J. Bolton (Eds.), Pattern Detection and Discovery. XII, 227 pages. 2002.

Vol. 2446: M. Klusch, S. Ossowski, O. Shehory (Eds.), Cooperative Information Agents VI. XI, 321 pages. 2002.

Vol. 2445: C. Anagnostopoulou, M. Ferrand, A. Smaill (Eds.), Music and Artificial Intelligence. VIII, 207 pages. 2002.

Vol. 2443: D. Scott (Ed.), Artificial Intelligence: Methodology, Systems, and Applications. X, 279 pages. 2002.

Vol. 2432: R. Bergmann, Experience Management. XXI, 393 pages. 2002.

Vol. 2431: T. Elomaa, H. Mannila, H. Toivonen (Eds.), Principles of Data Mining and Knowledge Discovery. XIV, 514 pages. 2002.

Vol. 2430: T. Elomaa, H. Mannila, H. Toivonen (Eds.), Machine Learning: ECML 2002. XIII, 532 pages. 2002.

Vol. 2427: M. Hannebauer, Autonomous Dynamic Reconfiguration in Multi-Agent Systems. XXI, 284 pages. 2002.

Vol. 2424: S. Flesca, S. Greco, N. Leone, G. Ianni (Eds.), Logics in Artificial Intelligence. XIII, 572 pages. 2002.

Table of Contents

Constraint Solving

Soft Constraints

Language Issues

Applications

A Comparative Study of Arithmetic Constraints on Integer Intervals

Krzysztof R. Apt[1,2] and Peter Zoeteweij[1]

[1] CWI, P.O. Box 94079, 1090 GB Amsterdam, the Netherlands
[2] University of Amsterdam, the Netherlands

Abstract. We propose here a number of approaches to implement constraint propagation for arithmetic constraints on integer intervals. To this end we introduce integer interval arithmetic. Each approach is explained using appropriate proof rules that reduce the variable domains. We compare these approaches using a set of benchmarks.

1 Preliminaries

1.1 Introduction

The subject of arithmetic constraints on reals has attracted a great deal of attention in the literature. For some reason arithmetic constraints on integer intervals have not been studied even though they are supported in a number of constraint programming systems. In fact, constraint propagation for them is present in ECL^iPS^e, SICStus Prolog, GNU Prolog, ILOG Solver and undoubtedly most of the systems that support constraint propagation for linear constraints on integer intervals. Yet, in contrast to the case of linear constraints — see notably [5] — we did not encounter in the literature any analysis of this form of constraint propagation.

In this paper we study these constraints in a systematic way. It turns out that in contrast to linear constraints on integer intervals there are a number of natural approaches to constraint propagation for these constraints.

To define them we introduce integer interval arithmetic that is modeled after the real interval arithmetic, see e.g., [6]. There are, however, essential differences since we deal with integers instead of reals. For example, multiplication of two integer intervals does not need to be an integer interval. In passing by we show that using integer interval arithmetic we can also define succinctly the well-known constraint propagation for linear constraints on integer intervals. In the second part of the paper we compare the proposed approaches by means of a set of benchmarks.

1.2 Constraint Satisfaction Problems

We review here the standard concepts of a constraint and of a constraint satisfaction problem. Consider a sequence of variables $X := x_1, \ldots, x_n$ where $n \geq 0$,

K.R. Apt et al. (Eds.): CSCLP 2003, LNAI 3010, pp. 1–24, 2004.

with respective domains $D_1, \ldots, D_n$ associated with them. So each variable x_i ranges over the domain D_i. By a ***constraint*** C on X we mean a subset of $D_1 \times \ldots \times D_n$. Given an element $d := d_1, \ldots, d_n$ of $D_1 \times \ldots \times D_n$ and a subsequence $Y := x_{i_1}, \ldots, x_{i_\ell}$ of X we denote by $d[Y]$ the sequence $d_{i_1}, \ldots, d_{i_\ell}$. In particular, for a variable x_i from X, $d[x_i]$ denotes d_i.

A ***constraint satisfaction problem***, in short CSP, consists of a finite sequence of variables X with respective domains $\mathcal{D}$, together with a finite set $\mathcal{C}$ of constraints, each on a subsequence of X. We write it as $\langle \mathcal{C} ; x_1 \in D_1, \ldots, x_n \in D_n \rangle$, where $X := x_1, \ldots, x_n$ and $\mathcal{D} := D_1, \ldots, D_n$.

By a ***solution*** to $\langle \mathcal{C} ; x_1 \in D_1, \ldots, x_n \in D_n \rangle$ we mean an element $d \in D_1 \times \ldots \times D_n$ such that for each constraint $C \in \mathcal{C}$ on a sequence of variables X we have $d[X] \in C$. We call a CSP ***consistent*** if it has a solution and ***inconsistent*** if it does not. Two CSPs with the same sequence of variables are called ***equivalent*** if they have the same set of solutions. In what follows we consider CSPs the constraints of which are defined in a simple language and identify the syntactic description of a constraint with its meaning being the set of tuples that satisfy it.

We view ***constraint propagation*** as a process of transforming CSPs that maintains their equivalence. In what follows we define this process by means of proof rules that act of CSPs and preserve equivalence. An interested reader can consult [1] for a precise explanation of this approach to describing constraint propagation.

1.3 Arithmetic Constraints

To define the arithmetic constraints use the alphabet that comprises

- variables,
- two constants, 0 and 1,
- the unary minus function symbol '$-$',
- three binary function symbols, '+','$-$'and '$\cdot$', all written in the infix notation.

By an ***arithmetic expression*** we mean a term formed in this alphabet and by an ***arithmetic constraint*** a formula of the form

$$s \; op \; t,$$

where s and t are arithmetic expressions and $op \in \{<, \leq, =, \neq, \geq, >\}$. For example

$$x^5 \cdot y^2 \cdot z^4 + 3x \cdot y^3 \cdot z^5 \leq 10 + 4x^4 \cdot y^6 \cdot z^2 - y^2 \cdot x^5 \cdot z^4 \tag{1}$$

is an arithmetic constraint. Here x^5 is an abbreviation for $x \cdot x \cdot x \cdot x \cdot x$ and similarly with the other expressions. If '$\cdot$' is not used in an arithmetic constraint, we call it a ***linear constraint***.

By an ***extended arithmetic expression*** we mean a term formed in the above alphabet extended by the unary function symbols '$.^n$' and '$\sqrt[n]{.}$' for each

$n \geq 1$ and the binary function symbol '/' written in the infix notation. For example

$$\sqrt[3]{(y^2 \cdot z^4)/(x^2 \cdot u^5)} \tag{2}$$

is an extended arithmetic expression. Here, in contrast to the above x^5 is a term obtained by applying the function symbol '$.^5$' to the variable x. The extended arithmetic expressions will be used only to define constraint propagation for the arithmetic constraints.

Fix now some arbitrary linear ordering $\prec$ on the variables of the language. By a ***monomial*** we mean an integer or a term of the form

$$a \cdot x_1^{n_1} \cdot \ldots \cdot x_k^{n_k}$$

where $k > 0$, $x_1, \ldots, x_k$ are different variables ordered w.r.t. $\prec$, and a is a non-zero integer and $n_1, \ldots, n_k$ are positive integers. We call then $x_1^{n_1} \cdot \ldots \cdot x_k^{n_k}$ the ***power product*** of this monomial.

Next, by a ***polynomial*** we mean a term of the form

$$\Sigma_{i=1}^{n} m_i,$$

where $n > 0$, at most one monomial m_i is an integer, and the power products of the monomials $m_1, \ldots, m_n$ are pairwise different. Finally, by a ***polynomial constraint*** we mean an arithmetic constraint of the form $s\ op\ b$, where s is a polynomial with no monomial being an integer, $op \in \{<, \leq, =, \neq, \geq, >\}$, and b is an integer. It is clear that by means of appropriate transformation rules we can transform each arithmetic constraint to a polynomial constraint. For example, assuming the ordering $x \prec y \prec z$ on the variables, the arithmetic constraint (1) can be transformed to the polynomial constraint

$$2x^5 \cdot y^2 \cdot z^4 - 4x^4 \cdot y^6 \cdot z^2 + 3x \cdot y^3 \cdot z^5 \leq 10$$

So, without loss of generality, from now on we shall limit our attention to the polynomial constraints.

Next, let us discuss the domains over which we interpret the arithmetic constraints. By an ***integer interval***, or an ***interval*** in short, we mean an expression of the form

$$[a..b]$$

where a and b are integers; $[a..b]$ denotes the set of all integers between a and b, including a and b. If $a > b$, we call $[a..b]$ the ***empty interval*** and denote it by $\emptyset$. Finally, by a ***range*** we mean an expression of the form

$$x \in I$$

where x is a variable and I is an interval.

2 Integer Set Arithmetic

To reason about the arithmetic constraints we employ a generalization of the arithmetic operations to the sets of integers.

2.1 Definitions

For X, Y sets of integers we define the following operations:

- addition:
$$X + Y := \{x + y \mid x \in X, y \in Y\},$$
- subtraction:
$$X - Y := \{x - y \mid x \in X, y \in Y\},$$
- multiplication:
$$X \cdot Y := \{x \cdot y \mid x \in X, y \in Y\},$$
- division:
$$X/Y := \{u \in \mathcal{Z} \mid \exists x \in X \exists y \in Y \; u \cdot y = x\},$$
- exponentiation:
$$X^n := \{x^n \mid x \in X\},$$
for each natural number $n > 0$,
- root extraction:
$$\sqrt[n]{X} := \{x \in \mathcal{Z} \mid x^n \in X\},$$
for each natural number $n > 0$.

All the operations except division are defined in the expected way. We shall return to it at the end of Section 6. At the moment it suffices to note the division operation is defined for all sets of integers, including $Y = \emptyset$ and $Y = \{0\}$. This division operation corresponds to the following division operation on the sets of reals introduced in [8]:

$$\frac{X}{Y} := \{u \in \mathcal{R} \mid \exists x \in X \exists y \in Y \; u \cdot y = x\}.$$

For a (n integer or real) number a and $op \in \{+, -, \cdot, /\}$ we identify $a \; op \; X$ with $\{a\} \; op \; X$ and $X \; op \; a$ with $X \; op \; \{a\}$.

To present the rules we are interested in we shall also use the addition and division operations on the sets of real numbers. Addition is defined in the same way as for the sets of integers, and division is defined above. In [6] it is explained how to implement these operations.

Further, given a set A of integers or reals, we define

$$^{\leq}A := \{x \in \mathcal{Z} \mid \exists a \in A \; x \leq a\},$$

$$^{\geq}A := \{x \in \mathcal{Z} \mid \exists a \in A \; x \geq a\}.$$

When limiting our attention to intervals of integers the following simple observation is of importance.

Note 1. For X, Y integer intervals and a an integer the following holds:

- $X \cap Y$, $X + Y, X - Y$ are integer intervals.
- $X/\{a\}$ is an integer interval.
- $X \cdot Y$ does not have to be an integer interval, even if $X = \{a\}$ or $Y = \{a\}$.
- X/Y does not have to be an integer interval.
- For each $n > 1$ X^n does not have to be an integer interval.
- For odd $n > 1$ $\sqrt[n]{X}$ is an integer interval.
- For even $n > 1$ $\sqrt[n]{X}$ is an integer interval or a disjoint union of two integer intervals. □

For example we have

$$[2..4] + [3..8] = [5..12],$$

$$[3..7] - [1..8] = [-5..6],$$

$$[3..3] \cdot [1..2] = \{3, 6\},$$

$$[3..5]/[-1..2] = \{-5, -4, -3, 2, 3, 4, 5\},$$

$$[-3..5]/[-1..2] = \mathcal{Z},$$

$$[1..2]^2 = \{1, 4\},$$

$$\sqrt[3]{[-30..100]} = [-3..4],$$

$$\sqrt[2]{[-100..9]} = [-3..3],$$

$$\sqrt[2]{[1..9]} = [-3..-1] \cup [1..3].$$

To deal with the problem that non-interval domains can be produced by some of the operations we introduce the following operation on the subsets of the set of the integers $\mathcal{Z}$:

$$int(X) := \begin{cases} \text{smallest integer interval containing } X & \text{if } X \text{ is finite,} \\ \mathcal{Z} & \text{otherwise.} \end{cases}$$

For example $int([3..5]/[-1..2]) = [-5..5]$ and $int([-3..5]/[-1..2]) = \mathcal{Z}$.

2.2 Implementation

To define constraint propagation for the arithmetic constraints on integer intervals we shall use the integer set arithmetic, mainly limited to the integer intervals. This brings us to the discussion of how to implement the introduced operations on the integer intervals. Since we are only interested in maintaining the property that the sets remain integer intervals or the set of integers $\mathcal{Z}$ we shall clarify how to implement the intersection, addition, subtraction and root extraction operations of the integer intervals and the $int(.)$ closure of the multiplication, division and exponentiation operations on the integer intervals. The case when one of the intervals is empty is easy to deal with. So we assume that we deal with non-empty intervals $[a..b]$ and $[c..d]$, that is $a \le b$ and $c \le d$.

Intersection, addition and subtraction. It is easy to see that

$$[a..b] \cap [c..d] = [max(a,c)..min(b,d)],$$
$$[a..b] + [c..d] = [a+c \,..\, b+d],$$
$$[a..b] - [c..d] = [a-d \,..\, b-c].$$

So the interval intersection, addition, and subtraction are straightforward to implement.

Root extraction. The outcome of the root extraction operator applied to an integer interval will be an integer interval or a disjoint union of two integer intervals. We shall explain in Section 4 why it is advantageous not to apply $int(.)$ to the outcome. This operator can be implemented by means of the following case analysis.

Case 1. Suppose n is odd. Then

$$\sqrt[n]{[a..b]} = [\lceil \sqrt[n]{a} \rceil \,..\, \lfloor \sqrt[n]{b} \rfloor].$$

Case 2. Suppose n is even and $b < 0$. Then

$$\sqrt[n]{[a..b]} = \emptyset.$$

Case 3. Suppose n is even and $b \geq 0$. Then

$$\sqrt[n]{[a..b]} = [-\lfloor |\sqrt[n]{b}| \rfloor \,..\, -\lceil |\sqrt[n]{a^+}| \rceil] \cup [\lceil |\sqrt[n]{a^+}| \rceil \,..\, \lfloor |\sqrt[n]{b}| \rfloor]$$

where $a^+ := max(0,a)$.

Multiplication. For the remaining operations we only need to explain how to implement the $int(.)$ closure of the outcome. First note that

$$int([a..b] \cdot [c..d]) = [min(A)..max(A)],$$

where $A = \{a \cdot c, a \cdot d, b \cdot c, b \cdot d\}$.

Using an appropriate case analysis we can actually compute the bounds of $int([a..b] \cdot [c..d])$ directly in terms of the bounds of the constituent intervals.

Division. In contrast, the $int(.)$ closure of the interval division is not so straightforward to compute. The reason is that, as we shall see in a moment, we cannot express the result in terms of some simple operations on the interval bounds.

Consider non-empty integer intervals $[a..b]$ and $[c..d]$. In analyzing the outcome of $int([a..b]/[c..d])$ we distinguish the following cases.

Case 1. Suppose $0 \in [a..b]$ and $0 \in [c..d]$.

Then by definition $int([a..b]/[c..d]) = \mathcal{Z}$. For example,

$$int([-1..100]/[-2..8]) = \mathcal{Z}.$$