Roman Barták
Michela Milano (Eds.)

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

LNCS 3524

Second International Conference, CPAIOR 2005
Prague, Czech Republic, May/June, 2005
Proceedings

Springer

Roman Barták Michela Milano (Eds.)

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

Second International Conference, CPAIOR 2005
Prague, Czech Republic, May 30–June 1, 2005
Proceedings

Springer

Volume Editors

Roman Barták
Charles University
Faculty of Mathematics and Physics
Malostranské nám. 2/25, 118 00 Prague 1, Czech Republic
E-mail: roman.bartak@mff.cuni.cz

Michela Milano
DEIS
University of Bologna
Viale Risorgimento 2, 40136 Bologna, Italy
E-mail: mmilano@deis.unibo.it

Library of Congress Control Number: 2005926642

CR Subject Classification (1998): G.1.6, G.1, G.2.1, F.2.2, I.2, J.1

ISSN 0302-9743
ISBN-10 3-540-26152-4 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-26152-0 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11493853 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3524

Lecture Notes in Computer Science

For information about Vols. 1–3419

please contact your bookseller or Springer

Vol. 3532: A. Gómez-Pérez, J. Euzenat (Eds.), The Semantic Web: Research and Applications. XV, 728 pages. 2005.

Vol. 3526: S.B. Cooper, B. Löwe, L. Torenvliet (Eds.), New Computational Paradigms. XVII, 574 pages. 2005.

Vol. 3525: A.E. Abdallah, C.B. Jones, J.W. Sanders (Eds.), Communicating Sequential Processes. XIV, 321 pages. 2005.

Vol. 3524: R. Barták, M. Milano (Eds.), Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems. XI, 411 pages. 2005.

Vol. 3523: J.S. Marques, N.P. de la Blanca, P. Pina (Eds.), Pattern Recognition and Image Analysis, Part II. XXVI, 733 pages. 2005.

Vol. 3522: J.S. Marques, N.P. de la Blanca, P. Pina (Eds.), Pattern Recognition and Image Analysis, Part I. XXVI, 703 pages. 2005.

Vol. 3520: O. Pastor, J. Falcão e Cunha (Eds.), Advanced Information Systems Engineering. XVI, 584 pages. 2005.

Vol. 3518: T.B. Ho, D. Cheung, H. Li (Eds.), Advances in Knowledge Discovery and Data Mining. XXI, 864 pages. 2005. (Subseries LNAI).

Vol. 3517: H.S. Baird, D.P. Lopresti (Eds.), Human Interactive Proofs. IX, 143 pages. 2005.

Vol. 3516: V.S. Sunderam, G.D.v. Albada, P.M.A. Sloot, J.J. Dongarra (Eds.), Computational Science – ICCS 2005, Part III. LXIII, 1143 pages. 2005.

Vol. 3515: V.S. Sunderam, G.D.v. Albada, P.M.A. Sloot, J.J. Dongarra (Eds.), Computational Science – ICCS 2005, Part II. LXIII, 1101 pages. 2005.

Vol. 3514: V.S. Sunderam, G.D.v. Albada, P.M.A. Sloot, J.J. Dongarra (Eds.), Computational Science – ICCS 2005, Part I. LXIII, 1089 pages. 2005.

Vol. 3513: A. Montoyo, R. Mu\~noz, E. Métais (Eds.), Natural Language Processing and Information Systems. XII, 408 pages. 2005.

Vol. 3510: T. Braun, G. Carle, Y. Koucheryavy, V. Tsaoussidis (Eds.), Wired/Wireless Internet Communications. XIV, 366 pages. 2005.

Vol. 3508: P. Bresciani, P. Giorgini, B. Henderson-Sellers, G. Low, M. Winikoff (Eds.), Agent-Oriented Information Systems II. X, 227 pages. 2005. (Subseries LNAI).

Vol. 3505: V. Gorodetsky, J. Liu, V. A. Skormin (Eds.), Autonomous Intelligent Systems: Agents and Data Mining. XIII, 303 pages. 2005. (Subseries LNAI).

Vol. 3503: S.E. Nikoletseas (Ed.), Experimental and Efficient Algorithms. XV, 624 pages. 2005.

Vol. 3502: F. Khendek, R. Dssouli (Eds.), Testing of Communicating Systems. X, 381 pages. 2005.

Vol. 3501: B. Kégl, G. Lapalme (Eds.), Advances in Artificial Intelligence. XV, 458 pages. 2005. (Subseries LNAI).

Vol. 3500: S. Miyano, J. Mesirov, S. Kasif, S. Istrail, P. Pevzner, M. Waterman (Eds.), Research in Computational Molecular Biology. XVII, 632 pages. 2005. (Subseries LNBI).

Vol. 3499: A. Pelc, M. Raynal (Eds.), Structural Information and Communication Complexity. X, 323 pages. 2005.

Vol. 3498: J. Wang, X. Liao, Z. Yi (Eds.), Advances in Neural Networks – ISNN 2005, Part III. L, 1077 pages. 2005.

Vol. 3497: J. Wang, X. Liao, Z. Yi (Eds.), Advances in Neural Networks – ISNN 2005, Part II. L, 947 pages. 2005.

Vol. 3496: J. Wang, X. Liao, Z. Yi (Eds.), Advances in Neural Networks – ISNN 2005, Part II. L, 1055 pages. 2005.

Vol. 3495: P. Kantor, G. Muresan, F. Roberts, D.D. Zeng, F.-Y. Wang, H. Chen, R.C. Merkle (Eds.), Intelligence and Security Informatics. XVIII, 674 pages. 2005.

Vol. 3494: R. Cramer (Ed.), Advances in Cryptology – EUROCRYPT 2005. XIV, 576 pages. 2005.

Vol. 3493: N. Fuhr, M. Lalmas, S. Malik, Z. Szlávik (Eds.), Advances in XML Information Retrieval. XI, 438 pages. 2005.

Vol. 3492: P. Blache, E. Stabler, J. Busquets, R. Moot (Eds.), Logical Aspects of Computational Linguistics. X, 363 pages. 2005. (Subseries LNAI).

Vol. 3489: G.T. Heineman, I. Crnkovic, H.W. Schmidt, J.A. Stafford, C. Szyperski, K. Wallnau (Eds.), Component-Based Software Engineering. XI, 358 pages. 2005.

Vol. 3488: M.-S. Hacid, N.V. Murray, Z.W. Raś, S. Tsumoto (Eds.), Foundations of Intelligent Systems. XIII, 700 pages. 2005. (Subseries LNAI).

Vol. 3486: T. Helleseth, D. Sarwate, H.-Y. Song, K. Yang (Eds.), Sequences and Their Applications - SETA 2004. XII, 451 pages. 2005.

Vol. 3483: O. Gervasi, M.L. Gavrilova, V. Kumar, A. Laganà, H.P. Lee, Y. Mun, D. Taniar, C.J.K. Tan (Eds.), Computational Science and Its Applications – ICCSA 2005, Part IV. XXVII, 1362 pages. 2005.

Vol. 3482: O. Gervasi, M.L. Gavrilova, V. Kumar, A. Laganà, H.P. Lee, Y. Mun, D. Taniar, C.J.K. Tan (Eds.), Computational Science and Its Applications – ICCSA 2005, Part III. LXVI, 1340 pages. 2005.

Vol. 3481: O. Gervasi, M.L. Gavrilova, V. Kumar, A. Laganà, H.P. Lee, Y. Mun, D. Taniar, C.J.K. Tan (Eds.), Computational Science and Its Applications – ICCSA 2005, Part II. LXIV, 1316 pages. 2005.

Vol. 3480: O. Gervasi, M.L. Gavrilova, V. Kumar, A. Laganà, H.P. Lee, Y. Mun, D. Taniar, C.J.K. Tan (Eds.), Computational Science and Its Applications – ICCSA 2005, Part I. LXV, 1234 pages. 2005.

Vol. 3479: T. Strang, C. Linnhoff-Popien (Eds.), Location- and Context-Awareness. XII, 378 pages. 2005.

Vol. 3478: C. Jermann, A. Neumaier, D. Sam (Eds.), Global Optimization and Constraint Satisfaction. XIII, 193 pages. 2005.

Vol. 3477: P. Herrmann, V. Issarny, S. Shiu (Eds.), Trust Management. XII, 426 pages. 2005.

Vol. 3475: N. Guelfi (Ed.), Rapid Integration of Software Engineering Techniques. X, 145 pages. 2005.

Vol. 3468: H.W. Gellersen, R. Want, A. Schmidt (Eds.), Pervasive Computing. XIII, 347 pages. 2005.

Vol. 3467: J. Giesl (Ed.), Term Rewriting and Applications. XIII, 517 pages. 2005.

Vol. 3465: M. Bernardo, A. Bogliolo (Eds.), Formal Methods for Mobile Computing. VII, 271 pages. 2005.

Vol. 3464: S.A. Brueckner, G.D.M. Serugendo, A. Karageorgos, R. Nagpal (Eds.), Engineering Self-Organising Systems. XIII, 299 pages. 2005. (Subseries LNAI).

Vol. 3463: M. Dal Cin, M. Kaâniche, A. Pataricza (Eds.), Dependable Computing - EDCC 2005. XVI, 472 pages. 2005.

Vol. 3462: R. Boutaba, K. Almeroth, R. Puigjaner, S. Shen, J.P. Black (Eds.), NETWORKING 2005. XXX, 1483 pages. 2005.

Vol. 3461: P. Urzyczyn (Ed.), Typed Lambda Calculi and Applications. XI, 433 pages. 2005.

Vol. 3460: Ö. Babaoglu, M. Jelasity, A. Montresor, C. Fetzer, S. Leonardi, A. van Moorsel, M. van Steen (Eds.), Self-star Properties in Complex Information Systems. IX, 447 pages. 2005.

Vol. 3459: R. Kimmel, N.A. Sochen, J. Weickert (Eds.), Scale Space and PDE Methods in Computer Vision. XI, 634 pages. 2005.

Vol. 3458: P. Herrero, M.S. Pérez, V. Robles (Eds.), Scientific Applications of Grid Computing. X, 208 pages. 2005.

Vol. 3456: H. Rust, Operational Semantics for Timed Systems. XII, 223 pages. 2005.

Vol. 3455: H. Treharne, S. King, M. Henson, S. Schneider (Eds.), ZB 2005: Formal Specification and Development in Z and B. XV, 493 pages. 2005.

Vol. 3454: J.-M. Jacquet, G.P. Picco (Eds.), Coordination Models and Languages. X, 299 pages. 2005.

Vol. 3453: L. Zhou, B.C. Ooi, X. Meng (Eds.), Database Systems for Advanced Applications. XXVII, 929 pages. 2005.

Vol. 3452: F. Baader, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning. XI, 562 pages. 2005. (Subseries LNAI).

Vol. 3450: D. Hutter, M. Ullmann (Eds.), Security in Pervasive Computing. XI, 239 pages. 2005.

Vol. 3449: F. Rothlauf, J. Branke, S. Cagnoni, D.W. Corne, R. Drechsler, Y. Jin, P. Machado, E. Marchiori, J. Romero, G.D. Smith, G. Squillero (Eds.), Applications of Evolutionary Computing. XX, 631 pages. 2005.

Vol. 3448: G.R. Raidl, J. Gottlieb (Eds.), Evolutionary Computation in Combinatorial Optimization. XI, 271 pages. 2005.

Vol. 3447: M. Keijzer, A. Tettamanzi, P. Collet, J.v. Hemert, M. Tomassini (Eds.), Genetic Programming. XIII, 382 pages. 2005.

Vol. 3444: M. Sagiv (Ed.), Programming Languages and Systems. XIII, 439 pages. 2005.

Vol. 3443: R. Bodik (Ed.), Compiler Construction. XI, 305 pages. 2005.

Vol. 3442: M. Cerioli (Ed.), Fundamental Approaches to Software Engineering. XIII, 373 pages. 2005.

Vol. 3441: V. Sassone (Ed.), Foundations of Software Science and Computational Structures. XVIII, 521 pages. 2005.

Vol. 3440: N. Halbwachs, L.D. Zuck (Eds.), Tools and Algorithms for the Construction and Analysis of Systems. XVII, 588 pages. 2005.

Vol. 3439: R.H. Deng, F. Bao, H. Pang, J. Zhou (Eds.), Information Security Practice and Experience. XII, 424 pages. 2005.

Vol. 3438: H. Christiansen, P.R. Skadhauge, J. Villadsen (Eds.), Constraint Solving and Language Processing. VIII, 205 pages. 2005. (Subseries LNAI).

Vol. 3437: T. Gschwind, C. Mascolo (Eds.), Software Engineering and Middleware. X, 245 pages. 2005.

Vol. 3436: B. Bouyssounouse, J. Sifakis (Eds.), Embedded Systems Design. XV, 492 pages. 2005.

Vol. 3434: L. Brun, M. Vento (Eds.), Graph-Based Representations in Pattern Recognition. XII, 384 pages. 2005.

Vol. 3433: S. Bhalla (Ed.), Databases in Networked Information Systems. VII, 319 pages. 2005.

Vol. 3432: M. Beigl, P. Lukowicz (Eds.), Systems Aspects in Organic and Pervasive Computing - ARCS 2005. X, 265 pages. 2005.

Vol. 3431: C. Dovrolis (Ed.), Passive and Active Network Measurement. XII, 374 pages. 2005.

Vol. 3430: S. Tsumoto, T. Yamaguchi, M. Numao, H. Motoda (Eds.), Active Mining. XII, 349 pages. 2005. (Subseries LNAI).

Vol. 3429: E. Andres, G. Damiand, P. Lienhardt (Eds.), Discrete Geometry for Computer Imagery. X, 428 pages. 2005.

Vol. 3428: Y.-J. Kwon, A. Bouju, C. Claramunt (Eds.), Web and Wireless Geographical Information Systems. XII, 255 pages. 2005.

Vol. 3427: G. Kotsis, O. Spaniol (Eds.), Wireless Systems and Mobility in Next Generation Internet. VIII, 249 pages. 2005.

Vol. 3423: J.L. Fiadeiro, P.D. Mosses, F. Orejas (Eds.), Recent Trends in Algebraic Development Techniques. VIII, 271 pages. 2005.

Vol. 3422: R.T. Mittermeir (Ed.), From Computer Literacy to Informatics Fundamentals. X, 203 pages. 2005.

Vol. 3421: P. Lorenz, P. Dini (Eds.), Networking - ICN 2005, Part II. XXXV, 1153 pages. 2005.

Vol. 3420: P. Lorenz, P. Dini (Eds.), Networking - ICN 2005, Part I. XXXV, 933 pages. 2005.

¥528.64元

Preface

The 2nd International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR 2005) was held in Prague, Czech Republic, during May 31–June 1, 2005.

The conference is intended primarily as a forum to focus on the integration and hybridization of the approaches of constraint programming (CP), artificial intelligence (AI), and operations research (OR) technologies for solving large-scale and complex real-life optimization problems. Therefore, CPAIOR is never far from industrial applications.

The high number of submissions received this year, almost 100 papers, in witness to the interest of the research community in this conference. From these submissions, we chose 26 to be published in full in the proceedings.

This volume includes summaries of the invited talks of CPAIOR: one from industry, one from the embedded system research community, and one from the operations research community. The invited speakers were: Filippo Focacci from ILOG S.A., France, one of the leading companies in the field; Paul Pop, professor in the Embedded Systems Lab in the Computer and Information Science Department, Linköping University; and Paul Williams, full professor of Operations Research at the London School of Economics.

The day before CPAIOR, a Master Class was organized by Gilles Pesant, where leading researchers gave introductory and overview talks in the area of metaheuristics and constraint programming. The Master Class was intended for PhD students, researchers, and practitioners. We are very grateful to Gilles who brought this excellent program together.

For conference publicity we warmly thank Willem Jan van Hoeve and Petr Vilím who did a great job with the high number of submissions received. We are very grateful to Michel Rueher who took care of the non-trivial task of finding funds for covering speakers' expenses, proceedings, and student grants.

Many thanks to the Program Committee, who reviewed all the submissions in detail and discussed conflicting papers deeply. Due to the unexpected number of submissions, their load was almost double that expected and their effort was repaid with nothing more than a free dinner.

A special thanks goes to Ondřej Čepek from Charles University and Milena Zeithamlová from Action M Agency who spent time in budgeting, planning, booking, and making it all work.

Finally, we would like to thank the sponsors who make it possible to organize this conference: the ARTIST Network of Excellence for sponsoring the talk by Paul Pop and making an interesting cross-fertilization possible; Carmen Systems, Sweden; CoLogNet, Network of Excellence; IISI (Intelligent Information Systems Institute, Cornell), USA; ILOG S.A., France; and SICS, Sweden.

June 2005 Roman Barták and Michela Milano

Organization

Organizers

Charles University in Prague, Faculty of Mathematics and Physics
Action M Agency (local arrangements)

Executive Committee

Roman Barták, Charles University, Czech Republic (conference co-chair)
Michela Milano, Università di Bologna, Italy (conference co-chair)
Ondřej Čepek, Charles University, Czech Republic (organization chair)

Program Committee

Abderrahmane Aggoun, Cosytec, France
Philippe Baptiste, Ecole Polytechnique, France
Roman Barták, Charles University, Czech Republic (chair)
Chris Beck, University of Toronto, Canada
Mats Carlsson, SICS, Sweden
Ondřej Čepek, Charles University, Czech Republic
Hani El Sakkout, CISCO, UK
Bernard Gendron, CRT and Univ. of Montreal, Canada
Carmen Gervet, IC-Parc, UK
Carla Gomes, Cornell University, USA
John Hooker, Carnegie Mellon University, USA
Narendra Jussien, Ecole des Mines de Nantes, France
Stefan Karisch, Carmen Systems, Canada
Francois Laburthe, Bouygues, France
Andrea Lodi, Università di Bologna, Italy
Michela Milano, Università di Bologna, Italy (chair)
George Nemhauser, Georgia Tech, USA
Gilles Pesant, CRT and Ecole Polytechnique de Montréal, Canada
Jean-Francois Puget, ILOG, France
Jean-Charles Régin, Cornell University, USA
Michel Rueher, University of Nice-Sophia Antipolis, France
Meinolf Sellmann, Brown University, USA
Helmut Simonis, IC-Parc, UK
Sven Thiel, Max Planck Institute, Germany

Gilles Trombettoni, University of Nice-Sophia Antipolis, France
Michael Trick, Carnegie Mellon University, USA
Pascal van Hentenryck, Brown University, USA
Mark Wallace, Monash University, Australia
Weixiong Zhang, Washington University, USA

Additional Referees

Carlos Ansotegui
Konstantin Artiouchine
Nicolas Beldiceanu
Hachemi Bennaceur
Thierry Benoist
Lucas Bordeaux
Ken Brown
Tom Carchrae
Alberto Caprara
Filipe Carvalho
David Daney
Pierre Deransart
Andrew Eremin
Xavier Gandibleux
Etienne Gaudin
Frédéric Goualard
Laurent Granvilliers
Jesper Hansen
Warwick Harvey
Vitaly Lagoon
Yahia Lebbah
Olivier Lhomme
Chu Min Li
Vassilis Liatsos
Tomas Liden
Ivana Ljubic
Ivan Luzzi
Roger Mailler
Michele Monaci
Bertrand Neveu
Stefano Novello
Ammar Oulamara
Nikos Papdakos
Thierry Petit
Ulrich Pferschy
Nikolai Pisaruk
Diego Fernandez Pons
Philippe Refalo
Guillaume Rochart
Andrea Roli
Benoit Rottembourg
Louis-Martin Rousseau
Jean-David Ruvini
Andrew Sadler
Paul Shaw
Barbara Smith
Stefano Smriglio
Peter Stuckey
Andrea Tramontani
Charlotte Truchet
Jean-Paul Watson
Quanshi Xia
Xiaolan Xie
Neil Yorke-Smith
Tallys Yunes
Alessandro Zanarini

Sponsors

ARTIST, Network of Excellence
Carmen Systems, Sweden
CoLogNet, Network of Excellence
IISI (Intelligent Information Systems Institute, Cornell), USA
ILOG S.A., France
SICS, Sweden

Table of Contents

Invited Papers

Technical Papers

Integration of Rules and Optimization in Plant PowerOps

Thomas Bousonville, Filippo Focacci, Claude Le Pape, Wim Nuijten, Frederic Paulin, Jean-Francois Puget, Anna Robert, and Alireza Sadeghin

ILOG S.A, 9 rue de Verdun, 94253 Gentilly, France
{tbousonville, ffocacci, clepape, wnuijten, fpaulin, jfpuget, anrobert, asadeghin}@ilog.fr

Abstract. Plant PowerOps (PPO) [9] is a new ILOG product, based on business rules and optimization technology, dedicated to production planning and detailed scheduling for manufacturing. This paper describes how PPO integrates a rule based system with the optimization engines and the graphical user interface. The integration proposed is motivated by the need to allow business users to manage unexpected changes in their environment. It provides a flexible interface for configuring, maintaining and tuning the system and for managing optimization scenarios. The proposed approach is discussed via several use cases we encountered in practice in supply chain management. Nevertheless, we believe that most of the ideas described in this paper apply in almost any area of optimization application.

1 Introduction

Most manufacturing companies are organized today around integrated programs called Enterprise Resource Planning (ERP) systems. ERP systems provide the information backbone needed to manage the day-to-day execution handling the many transactions that document the activity of a company. Since the beginning of the new century, Advanced Planning and Scheduling (APS) systems have been increasingly adopted to plan the production taking into account capacity and material flow constraints in order to meet customer demand. APS systems embed algorithms for planning and scheduling spanning from the application of very simple priority rules to complex optimization algorithms depending on the needs of each customer. Although rule-based scheduling and simulation-based scheduling are still widely used, today the best APS systems offer scheduling algorithms based on Meta-heuristics, Constraint Programming and Mathematical Programming.

The highly competitive marketplace on the one hand pushes to improve the production efficiency; on the other hand it pushes to increase the flexibility necessary to adapt to the continuous variations of customer demand. Today manufacturing companies need to produce a higher variety of products and customized products. The increasing needs for flexibility are pushing today's APS

R. Barták and M. Milano (Eds.): CPAIOR 2005, LNCS 3524, pp. 1–15, 2005.

systems to their limits. Many companies are struggling with the limitation of the first generation of APS systems and are looking for new solutions.

A company that needs to implement advanced supply chain optimization tools has two possible choices: either it will implement an APS package or it will build it using optimization components and technology via often long and costly custom development. The drawback of buying an existing APS package is that it provides a generic optimization model which will not take into consideration all production constraints and policies characteristic of the company. Often the company is forced to fit into the predefined model. The bottom line can be a very high total cost of ownership combined with unhappy end users who, in some cases, replace the system with their previously developed Excel spreadsheet. The alternative of developing a custom solution is only viable for few companies (often with large OR departments). And even in this case the usability of custom development is not guaranteed. In both cases, changing the supply chain optimization system to follow the rapidly evolving business conditions is an issue.

The challenge for APS packages vendors is therefore to provide enough generality to avoid developing an optimization engine for each and every customer and to build a flexible and configurable enough system to meet the real needs of the customer. Ideally, such package should be configurable by its business users. These are the people that actually solve a business problem, such as producing the production plan for a plant. These business users usually do not possess the IT skills that are needed for adapting an APS package to the peculiarities of their plant.

There are many ways in which an IT package could be made more flexible. One could add a scripting language to it for instance. Unfortunately, even scripting languages are deemed to be too complex to be learned and used by business users. Another possibility could be to use tools that business users use, such as a spreadsheet. However, a spreadsheet interface is not powerful enough to express complex use cases such as business policies. A third approach to flexibility is emerging nowadays. It is called business rules. This approach let business users make statements about their business in a friendly way. These rules are then used to preprocess (or postprocess) the input (or output) of an optimization application. This use of rules is quite different than the so called rule inference systems that were used in expert systems in the 80's. Indeed, rules aren't used here to solve a problem. Rather, they are used to state what the problem is really about. This difference is at the root of the current success of business rules in the market place.

We are convinced that advances and the increased popularity of Business Rules create the opportunity to provide the flexibility the lack of which was limiting the applicability of APS systems. In this paper we describe how Plant PowerOps [9] takes advantage of this technology and we claim that the proposed rules interface can be generalized to the vast majority of optimization applications. Indeed, although this paper does not propose any advance in either rule based systems or operations research, it presents a new, extremely pragmatic, way of applying and integrating rule based systems to optimization models and algorithms.

The structure of the paper is as follows: in section 2 we present an overview of Plant PowerOps briefly describing the types of problems solved by PPO and its architecture. Section 3 is devoted to present different use cases where the integration of rules and optimization is demonstrated to be a powerful combination to overcome the limits of today's optimization software. Section 4 explains the reasons behind some of the design decisions taken in the development of the proposed integration, discusses open questions and future work. Section 5 presents some related approaches. Section 6 concludes the paper.

2 Plant PowerOps Overview

Plant PowerOps (PPO) is a new *Advanced Planning and Scheduling (APS)* system dedicated to production planning and detailed scheduling for manufacturing. It enables users to plan the production taking into account capacity and material flow constraints to meet customer demand.

Although Plant PowerOps provides production planning, lot sizing, and detailed scheduling engines, due to space limitations, we will concentrate the examples to the detailed scheduling features of PPO. The scheduling engine is used to schedule production activities (such as chemical reactions, mixing, forming, assembling or separating), and setup activities (such as cleaning or preparing) on different machines or production lines in order to efficiently produce quality finished products in a timely manner, while satisfying customer demand for the finished product.

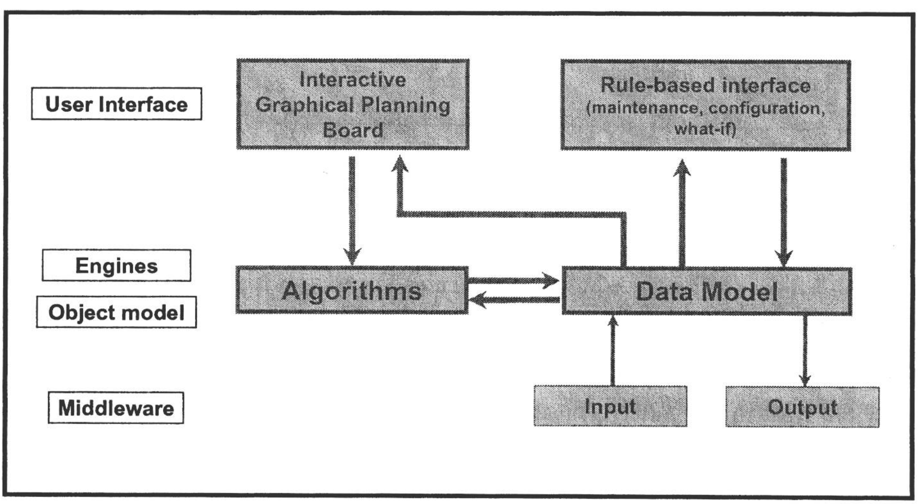

Fig. 1. Architecture

As shown in figure 1, Plant PowerOps provides

- a pre-defined data model to capture intricacies of the manufacturing operations. The pre-defined data model represents, for example, production recipes, production orders, calendars (e.g. breaks, shifts, productivity profiles), resources, customer demands.
- effective optimization models and algorithms based on Mathematical Programming, Constraint Programming and Local Search. These algorithms automatically generate feasible, cost-effective detailed schedules minimizing a combination of objective functions such as total tardiness, total earliness, total setup cost, makespan, processing cost, etc.
- a graphical planning board to visualize, analyze, manually adjust and update production schedules.
- a rule-based customization interface to configure and maintain over time the graphical user interface and the parameters of the model and of the algorithm. The rule-based interface provides also the ability to define optimization scenarios and to modify problem data and solutions.
- an integration framework to connect PPO to a database, to an existing ERP (Enterprise Resource Planning), or to an existing MES (Manufacturing Execution System).

2.1 Rule-Based Interface

The integration between the business rules and the optimization model and between the business rules and the graphical user interface is loose. Rules apply either in a pre-processing step (i.e. before the execution of the engine) or in a post-processing step (i.e. after the execution of the engine).

The rule interface of PPO is based on *ILOG JRules* [8] which parses and interprets *production rules* and executes them in forward-chaining using the Rete algorithm [5].

The typical syntax of the production rules interpreted by PPO is the following:

```
when
    conditions
then
    actions
```

Conditions (or left hand side of the rule) are methods returning booleans on objects of the PPO data model. For example, a condition looking for all activities which are due on Jan 1st is translated as *evaluate(theActivity.getDueDate() equals "Jan 1, 2005")*. *Actions* (or right hand side of the rule) can be (i) any method (returning void) typically modifying the state of the objects, (ii) insertion in the working memory, (iii) retraction from working memory. We use actions to produce side effects on the model or on the solution. On top of the rule language, JRules provides a syntax in natural language like flavor that is used in all the examples shown in this paper.

The rule engine and the optimization engine are fully independent and communicate only through modification of the model and of the solutions. The Plant PowerOps user selects the rules she/he wants to apply in a given scenario before or after the optimization per se. Moreover, the optimization model is exposed via a high level *closed* interface. The interface is *closed* in the sense that we do not provide direct access to decision variables so that only predefined constraints are possible.

The left hand side of a rule checks conditions on the model and its right hand side produces side effects on the model if the conditions are met. Rules apply either in a pre-processing step (i.e. before the execution of the engine) or in a post-processing step (i.e. after the execution of the engine). Typically, a pre-processing rule can be seen as a way to transform a model (coming from the legacy system) into a new model upon which optimization is performed. The optimization engine optimizes the transformed model and has no knowledge of the rules that applied to generate it. Post-processing rules check the state of the model and the solutions after optimization has occurred and possibly modify the solutions. The advantage of the loose integration proposed relies on its simplicity and modularity.

Note that the way rules are used in PPO is very different from the way rules are used in expert systems. In expert systems rules are used to solve the problem at hand. This usually requires complex sequences of rules firing, and maintenance was a real concern. In PPO the number of rules we expect to be active is very limited; the rules are often independent from each other and rarely chained together. This simplifies the maintenance issue while still allowing business users to understand and manage them.

3 Use Cases

3.1 The Chocolate Factory

In order to demonstrate the interest of integrating business rules and optimization algorithms we will describe some use cases that could be faced by supply chain managers and production planners of an imaginary chocolate factory.

This imaginary factory produces chocolate confections. Many production steps and machines are required to complete the manufacturing process. The manufacturing process is driven by customer demands and production orders, these processes being driven by recipes and the materials they produce. Costly setup times are required, and multiple process modes (e.g. an activity may be performed in alterative machines) are possible and may be associated with different process costs. Also, activities have precedence constraints.

The factory produces chocolate eggs, rabbits and squirrels. These can be made of either dark chocolate or milk chocolate. Each shape of product –egg (E), rabbit (R), or squirrel (S)– made of either dark (D) or milk (M) chocolate, can be filled with coconut cream (C), hazelnut cream (H), or filled with nothing (N). The possible combinations are identified using three-letter acronym such