Jean-Charles Régin
Michel Rueher (Eds.)

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

LNCS 3011

First International Conference, CPAIOR 2004
Nice, France, April 2004
Proceedings

Springer

Jean-Charles Régin Michel Rueher (Eds.)

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

First International Conference, CPAIOR 2004
Nice, France, April 20-22, 2004
Proceedings



Springer

Volume Editors

Jean-Charles Régin
ILOG
Les Taissounières HB2, 1681 route des Dolines, 06560 Valbonne, France
E-mail: regin@ilog.fr

Michel Rueher
Université de Nice - Sophia Antipolis
Projet COPRIN, 13S/CNRS/INRIA, ESSI
BP 145, 06903 Sophia Antipolis, France
E-mail: Michel.Rueher@sophia.inria.fr

Library of Congress Control Number: 2004103824

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

ISSN 0302-9743
ISBN 3-540-21836-X Springer-Verlag Berlin Heidelberg New York



Typesetting: Camera-ready by author, data conversion by DA-TeX Gerd Blumenstein
Printed on acid-free paper SPIN: 10997383 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3011

*Commenced Publication in 1973*
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Springer
*Berlin*
*Heidelberg*
*New York*
*Hong Kong*
*London*
*Milan*
*Paris*
*Tokyo*

# Preface

This volume contains the proceedings of the 1st International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimisation Problems. This new conference follows the series of CP-AI-OR International Workshops on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimisation Problems held in Ferrara (1999), Paderborn (2000), Ashford (2001), Le Croisic (2002), and Montreal (2003). The success of the previous workshops has demonstrated that CP-AI-OR is becoming a major forum for exchanging ideas and methodologies from both fields. The aim of this new conference is to bring together researchers from AI and OR, and to give them the opportunity to show how the integration of techniques from AI and OR can lead to interesting results on large-scale and complex problems.

The integration of techniques from artificial intelligence and operations research has provided effective algorithms for tackling complex and large-scale combinatorial problems with significant improvements in terms of efficiency, scalability and optimality. The benefit of this integration has been shown in applications such as hoist scheduling, rostering, dynamic scheduling and vehicle routing. At the programming and modelling levels, most constraint languages embed OR techniques to reason about collections of constraints, so-called global constraints. Some languages also provide support for hybridization allowing the programmer to build new integrated algorithms. The resulting multi paradigm programming framework combines the flexibility and modelling facilities of constraint programming with the special purpose and efficient methods from operations research.

CP-AI-OR 2004 was intended primarily as a forum to focus on the integration of the approaches of CP, AI and OR technologies. A secondary aim was to provide an opportunity for researchers in one area to learn about techniques in the others. Fifty-six papers were submitted in response to the call for papers. After the reviewing period and some online discussions, the program committee met physically at Nice on January 30 and 31, 2004. The program committee decided to accept 23 technical papers and 7 short papers. Short papers present interesting recent results or novel thought-provoking ideas that are not quite ready for a regular full-length paper. Both types of papers were reviewed rigorously and held to a very high standard.

CP-AI-OR 2004 was fortunate to attract outstanding invited talks. Heinrich Braun and Thomas Kasper discussed the challenges of optimization problems in supply chain management. Ignacio Grossmann proposed a hybrid framework that uses mathematical and constraint programming for the scheduling of batch chemical processes. Michel Minoux told us about strengthened relaxations for some CP-resistant combinatorial problems and their potential usefulness.

We wish to thank our generous sponsors who allowed us to offer substantial allowances to students attending the conference in order to cover their expenses.

We extend our gratitude to the outstanding program committee who worked very hard under tight deadlines. We are deeply grateful to Claude Michel who worked in the trenches in preparing the CP meeting at Nice and who dealt with all the difficult organization aspects of this conference.

April 2004 Jean-Charles Régin and Michel Rueher

# Organization

CPAIOR 2004 was organized by INRIA (Institut National de Recherche en Informatique et en Automatique) Sophia Antipolis.

## Executive Committee

| | |
|---|---|
| Conference and Program Chairs | Jean-Charles Régin (ILOG S.A.) and<br>Michel Rueher (Univ. of Nice-Sophia Antipolis) |
| Organization Chair | Claude Michel (Univ. of Nice-Sophia Antipolis) |

## Program Committee

Abderrahmane Aggoun, Cosytec, France
Philippe Baptiste, École Polytechnique, France
Roman Bartak, Charles University, Czech Republic
Chris Beck, Cork Constraint Computation Centre, Ireland
Mats Carlsson, SICS, Sweden
Alain Colmerauer, Univ. of Marseille, France
Hani El Sakkout, Parc Technologies, UK
Bernard Gendron, CRT and Univ. of Montreal, Canada
Carmen Gervet, IC-Parc, UK
Carla Gomes, Cornell University, USA
Narendra Jussien, École des Mines de Nantes, France
Stefan Karisch, Carmen Systems, Canada
François Laburthe, Bouygues, France
Olivier Lhomme, ILOG, France
Michela Milano, Univ. of Bologna, Italy
George Nemhauser, Georgia Tech, USA
Gilles Pesant, CRT and École Polytechnique de Montreal, Canada
Jean-Charles Régin (chair), ILOG, France
Michel Rueher (chair), Univ. of Nice-Sophia Antipolis, France
Christian Schulte, KTH, Sweden
Meinolf Sellmann, Cornell University, USA
Sven Thiel, Max Planck Institute, Germany
Gilles Trombettoni, Univ. of Nice-Sophia Antipolis, France
Michael Trick, Carnegie Mellon University, USA
Pascal van Hentenryck, Brown University, USA
Mark Wallace, IC-Parc, UK

## Referees

Abderrahmane Aggoun
Philippe Baptiste
Roman Bartak
Chris Beck
Nicolas Beldiceanu
Pascal Brisset
Mats Carlsson
Alain Colmerauer
Miguel Constantino
Romuald Debruyne
Hani El Sakkout
Andrew Eremin
Marco Gavanelli
Bernard Gendron
Carmen Gervet
Carla Gomes
Idir Gouachi
Brahim Hnich
Narendra Jussien
Stefan Karisch
Irit Katriel
Yahia Lebbah
François Laburthe
Olivier Lhomme
Vassilis Liatsos
Andrea Lodi
Michela Milano
George Nemhauser
Bertrand Neveu
Stefano Novello
Gilles Pesant
Nikolai Pisaruk
Steven Prestwich
Philippe Refalo
Guillaume Rochart
Louis-Martin Rousseau
Christian Schulte
Meinolf Sellmann
Paul Shaw
Josh Singer
Helmut Simonis
Neil Yorke-Smith
Francis Sourd
Sven Thiel
Gilles Trombettoni
Michael Trick
Pascal van Hentenryck
Willem Jan van Hoeve
Mark Wallace
Jean-Paul Watson

## Sponsors

Bouygues, France
Cosytec S.A., France
Carmen System
CoLogNET
ESSI (École Supérieure en Sciences Informatiques), Sophia Antipolis, France
I3S/CNRS–Université de Nice-Sophia Antipolis, France
IISI (Intelligent Information Systems Institute), USA
ILOG S.A., Paris

# Lecture Notes in Computer Science

For information about Vols. 1–2896

please contact your bookseller or Springer-Verlag

Vol. 3025: G.A. Vouros, T. Panayiotopoulos (Eds.), Methods and Applications of Artificial Intelligence. XV, 546 pages. 2004. (Subseries LNAI).

Vol. 3015: C. Barakat, I. Pratt (Eds.), Passive and Active Network Measurement. XI, 300 pages. 2004.

Vol. 3011: J.-C. Régin, M. Rueher (Eds.), Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems. XI, 415 pages. 2004.

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

Vol. 3009: F. Bomarius, H. Iida (Eds.), Product Focused Software Process Improvement. XIV, 584 pages. 2004.

Vol. 3007: J.X. Yu, X. Lin, H. Lu, Y. Zhang (Eds.), Advanced Web Technologies and Applications. XXII, 936 pages. 2004.

Vol. 3006: M. Matsui, R. Zuccherato (Eds.), Selected Areas in Cryptography. XI, 361 pages. 2004.

Vol. 3005: G.R. Raidl, S. Cagnoni, J. Branke, D.W. Corne, R. Drechsler, Y. Jin, C.G. Johnson, P. Machado, E. Marchiori, F. Rothlauf, G.D. Smith, G. Squillero (Eds.), Applications of Evolutionary Computing. XVII, 562 pages. 2004.

Vol. 3004: J. Gottlieb, G.R. Raidl (Eds.), Evolutionary Computation in Combinatorial Optimization. X, 241 pages. 2004.

Vol. 3003: M. Keijzer, U.-M. O'Reilly, S.M. Lucas, E. Costa, T. Soule (Eds.), Genetic Programming. XI, 410 pages. 2004.

Vol. 3001: A. Ferscha, F. Mattern (Eds.), Pervasive Computing. XIII, 358 pages. 2004.

Vol. 2999: E.A. Boiten, J. Derrick, G. Smith (Eds.), Integrated Formal Methods. XI, 541 pages. 2004.

Vol. 2998: Y. Kameyama, P.J. Stuckey (Eds.), Functional and Logic Programming. X, 307 pages. 2004.

Vol. 2997: S. McDonald, J. Tait (Eds.), Advances in Information Retrieval. XIII, 427 pages. 2004.

Vol. 2996: V. Diekert, M. Habib (Eds.), STACS 2004. XVI, 658 pages. 2004.

Vol. 2995: C. Jensen, S. Poslad, T. Dimitrakos (Eds.), Trust Management. XIII, 377 pages. 2004.

Vol. 2994: E. Rahm (Ed.), Data Integration in the Life Sciences. X, 221 pages. 2004. (Subseries LNBI).

Vol. 2993: R. Alur, G.J. Pappas (Eds.), Hybrid Systems: Computation and Control. XII, 674 pages. 2004.

Vol. 2992: E. Bertino, S. Christodoulakis, D. Plexousakis, V. Christophides, M. Koubarakis, K. Böhm, E. Ferrari (Eds.), Advances in Database Technology - EDBT 2004. XVIII, 877 pages. 2004.

Vol. 2991: R. Alt, A. Frommer, R.B. Kearfott, W. Luther (Eds.), Numerical Software with Result Verification. X, 315 pages. 2004.

Vol. 2989: S. Graf, L. Mounier (Eds.), Model Checking Software. X, 309 pages. 2004.

Vol. 2988: K. Jensen, A. Podelski (Eds.), Tools and Algorithms for the Construction and Analysis of Systems. XIV, 608 pages. 2004.

Vol. 2987: I. Walukiewicz (Ed.), Foundations of Software Science and Computation Structures. XIII, 529 pages. 2004.

Vol. 2986: D. Schmidt (Ed.), Programming Languages and Systems. XII, 417 pages. 2004.

Vol. 2985: E. Duesterwald (Ed.), Compiler Construction. X, 313 pages. 2004.

Vol. 2984: M. Wermelinger, T. Margaria-Steffen (Eds.), Fundamental Approaches to Software Engineering. XII, 389 pages. 2004.

Vol. 2983: S. Istrail, M.S. Waterman, A. Clark (Eds.), Computational Methods for SNPs and Haplotype Inference. IX, 153 pages. 2004. (Subseries LNBI).

Vol. 2982: N. Wakamiya, M. Solarski, J. Sterbenz (Eds.), Active Networks. XI, 308 pages. 2004.

Vol. 2981: C. Müller-Schloer, T. Ungerer, B. Bauer (Eds.), Organic and Pervasive Computing – ARCS 2004. XI, 339 pages. 2004.

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

Vol. 2978: R. Groz, R.M. Hierons (Eds.), Testing of Communicating Systems. XII, 225 pages. 2004.

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

Vol. 2976: M. Farach-Colton (Ed.), LATIN 2004: Theoretical Informatics. XV, 626 pages. 2004.

Vol. 2973: Y. Lee, J. Li, K.-Y. Whang, D. Lee (Eds.), Database Systems for Advanced Applications. XXIV, 925 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. (Subseries LNAI).

Vol. 2971: J.I. Lim, D.H. Lee (Eds.), Information Security and Cryptology -ICISC 2003. XI, 458 pages. 2004.

Vol. 2970: F. Fernández Rivera, M. Bubak, A. Gómez Tato, R. Doallo (Eds.), Grid Computing. XI, 328 pages. 2004.

Vol. 2964: T. Okamoto (Ed.), Topics in Cryptology – CT-RSA 2004. XI, 387 pages. 2004.

Vol. 2963: R. Sharp, Higher Level Hardware Synthesis. XVI, 195 pages. 2004.

Vol. 2962: S. Bistarelli, Semirings for Soft Constraint Solving and Programming. XII, 279 pages. 2004.

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

Vol. 2960: P.D. Mosses (Ed.), CASL Reference Manual. XVII, 528 pages. 2004.

Vol. 2958: L. Rauchwerger (Ed.), Languages and Compilers for Parallel Computing. XI, 556 pages. 2004.

Vol. 2957: P. Langendoerfer, M. Liu, I. Matta, V. Tsaoussidis (Eds.), Wired/Wireless Internet Communications. XI, 307 pages. 2004.

Vol. 2954: F. Crestani, M. Dunlop, S. Mizzaro (Eds.), Mobile and Ubiquitous Information Access. X, 299 pages. 2004.

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

Vol. 2952: N. Guelfi, E. Astesiano, G. Reggio (Eds.), Scientific Engineering of Distributed Java Applications. X, 157 pages. 2004.

Vol. 2951: M. Naor (Ed.), Theory of Cryptography. XI, 523 pages. 2004.

Vol. 2949: R. De Nicola, G. Ferrari, G. Meredith (Eds.), Coordination Models and Languages. X, 323 pages. 2004.

Vol. 2948: G.L. Mullen, A. Poli, H. Stichtenoth (Eds.), Finite Fields and Applications. VIII, 263 pages. 2004.

Vol. 2947: F. Bao, R. Deng, J. Zhou (Eds.), Public Key Cryptography – PKC 2004. XI, 455 pages. 2004.

Vol. 2946: R. Focardi, R. Gorrieri (Eds.), Foundations of Security Analysis and Design II. VII, 267 pages. 2004.

Vol. 2943: J. Chen, J. Reif (Eds.), DNA Computing. X, 225 pages. 2004.

Vol. 2941: M. Wirsing, A. Knapp, S. Balsamo (Eds.), Radical Innovations of Software and Systems Engineering in the Future. X, 359 pages. 2004.

Vol. 2940: C. Lucena, A. Garcia, A. Romanovsky, J. Castro, P.S. Alencar (Eds.), Software Engineering for Multi-Agent Systems II. XII, 279 pages. 2004.

Vol. 2939: T. Kalker, I.J. Cox, Y.M. Ro (Eds.), Digital Watermarking. XII, 602 pages. 2004.

Vol. 2937: B. Steffen, G. Levi (Eds.), Verification, Model Checking, and Abstract Interpretation. XI, 325 pages. 2004.

Vol. 2936: P. Liardet, P. Collet, C. Fonlupt, E. Lutton, M. Schoenauer (Eds.), Artificial Evolution. XIV, 410 pages. 2004.

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

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

Vol. 2929: H. de Swart, E. Orlowska, G. Schmidt, M. Roubens (Eds.), Theory and Applications of Relational Structures as Knowledge Instruments. VII, 273 pages. 2003.

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

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

Vol. 2919: E. Giunchiglia, A. Tacchella (Eds.), Theory and Applications of Satisfiability Testing. XI, 530 pages. 2004.

Vol. 2917: E. Quintarelli, Model-Checking Based Data Retrieval. XVI, 134 pages. 2004.

Vol. 2916: C. Palamidessi (Ed.), Logic Programming. XII, 520 pages. 2003.

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

Vol. 2914: P.K. Pandya, J. Radhakrishnan (Eds.), FST TCS 2003: Foundations of Software Technology and Theoretical Computer Science. XIII, 446 pages. 2003.

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

Vol. 2911: T.M.T. Sembok, H.B. Zaman, H. Chen, S.R. Urs, S.H. Myaeng (Eds.), Digital Libraries: Technology and Management of Indigenous Knowledge for Global Access. XX, 703 pages. 2003.

Vol. 2910: M.E. Orlowska, S. Weerawarana, M.M.P. Papazoglou, J. Yang (Eds.), Service-Oriented Computing - ICSOC 2003. XIV, 576 pages. 2003.

Vol. 2909: R. Solis-Oba, K. Jansen (Eds.), Approximation and Online Algorithms. VIII, 269 pages. 2004.

Vol. 2908: K. Chae, M. Yung (Eds.), Information Security Applications. XII, 506 pages. 2004.

Vol. 2907: I. Lirkov, S. Margenov, J. Wasniewski, P. Yalamov (Eds.), Large-Scale Scientific Computing. XI, 490 pages. 2004.

Vol. 2906: T. Ibaraki, N. Katoh, H. Ono (Eds.), Algorithms and Computation. XVII, 748 pages. 2003.

Vol. 2905: A. Sanfeliu, J. Ruiz-Shulcloper (Eds.), Progress in Pattern Recognition, Speech and Image Analysis. XVII, 693 pages. 2003.

Vol. 2904: T. Johansson, S. Maitra (Eds.), Progress in Cryptology - INDOCRYPT 2003. XI, 431 pages. 2003.

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

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

Vol. 2901: F. Bry, N. Henze, J. Ma luszyński (Eds.), Principles and Practice of Semantic Web Reasoning. X, 209 pages. 2003.

Vol. 2900: M. Bidoit, P.D. Mosses (Eds.), Casl User Manual. XIII, 240 pages. 2004.

Vol. 2899: G. Ventre, R. Canonico (Eds.), Interactive Multimedia on Next Generation Networks. XIV, 420 pages. 2003.

Vol. 2898: K.G. Paterson (Ed.), Cryptography and Coding. IX, 385 pages. 2003.

Vol. 2897: O. Balet, G. Subsol, P. Torguet (Eds.), Virtual Storytelling. XI, 240 pages. 2003.

# Table of Contents

## Invited Paper



## Technical Papers

## Short Papers

# Using MILP and CP for the Scheduling of Batch Chemical Processes

Christos T. Maravelias and Ignacio E. Grossmann

Department of Chemical Engineering
Carnegie Mellon University, Pittsburgh, PA 15213, USA
{ctm,ig0c}@andrew.cmu.edu

**Abstract.** A hybrid framework that uses Mathematical and Constraint Programming for the scheduling of batch chemical processes is proposed. Mathematical programming is used for the high-level optimization decisions (number and type of tasks, and assignment of equipment units to tasks), and Constraint Programming is used for the low-level sequencing decisions. The original problem is decomposed into an MILP master problem and a CP subproblem. The master MILP is a relaxation of the original problem, and given a relaxed solution, the CP subproblem checks whether there is a feasible solution and generates integer cuts. The proposed framework is based on the hybrid algorithm of Maravelias and Grossmann ([1],[2]), and can be used for different objective functions and different plant configurations. In this paper we present the simplifications and enhancements that allow us to use the proposed framework in a variety of problems, and report computational results.

## 1 Introduction

Scheduling of operations is a common and very important problem in the chemical industry. While related problems have been extensively studied in the Operations Research literature (see [3]), this area has only been addressed recently in process systems engineering (see [4], [5], [6], [7] for reviews).

In terms of plant configurations, problems in chemical industry can be classified into four major categories. In multiple-unit or single-stage plants (Figure 1.a), there are $N$ orders to be scheduled in $M$ units. In flow-shop multi-stage plants (Figure 1.b), there are $N$ orders to be processed in $K$ stages, following the same order. Each stage $k \in \{1,...K\}$ consists of $M_k$ units, and each order must be processed by one unit in each stage. In general multi-stage plants (Figure 1.c), each order must be processed in all stages but not in the same order. Multipurpose batch plants (Figure 1.d), finally, can be viewed as a generalization of all previous configurations, where batch splitting and mixing, as well as recycle streams are present. It should be noted that the original work in process scheduling concentrated mostly on flow-shop and general multi-stage batch plants (see [3], [5]). The study of general multipurpose plants was largely promoted by the work of Kondili et al. [8].

Preemption is usually not allowed, and utility constraints (e.g. manpower, cooling water, etc.), and release/due times may be present in all configurations. Different storage policies, such as unlimited intermediate storage (UIS), finite intermediate

J.-C. Régin and M. Rueher (Eds.): CPAIOR 2004, LNCS 3011, pp. 1-20, 2004.

storage (FIS), no intermediate storage (NIS) and zero-wait (ZW), are used in multi-purpose and multi-stage plants, while in singe-stage plants it is usually assumed that unlimited storage is available. Furthermore, the batch size of a task may be variable, which in many cases leads to variable duration and utility requirements. In terms of objective functions, the most common ones: maximization of production over a fixed time horizon, minimization of makespan for given demand and minimization of cost for given demand with due dates.

A feature that makes scheduling problems in chemical industry hard to solve is that usually the type and number of tasks (jobs) are not uniquely defined, and moreover, a specific task can be performed in more than one unit. These problems are hard to solve because of the large number of different solutions, and have not been studied extensively. Problems where the number and type of tasks are fixed and each task can be assigned to only one machine, on the other hand, have been extensively studied in OR community and efficient algorithms exist for many of these problems.

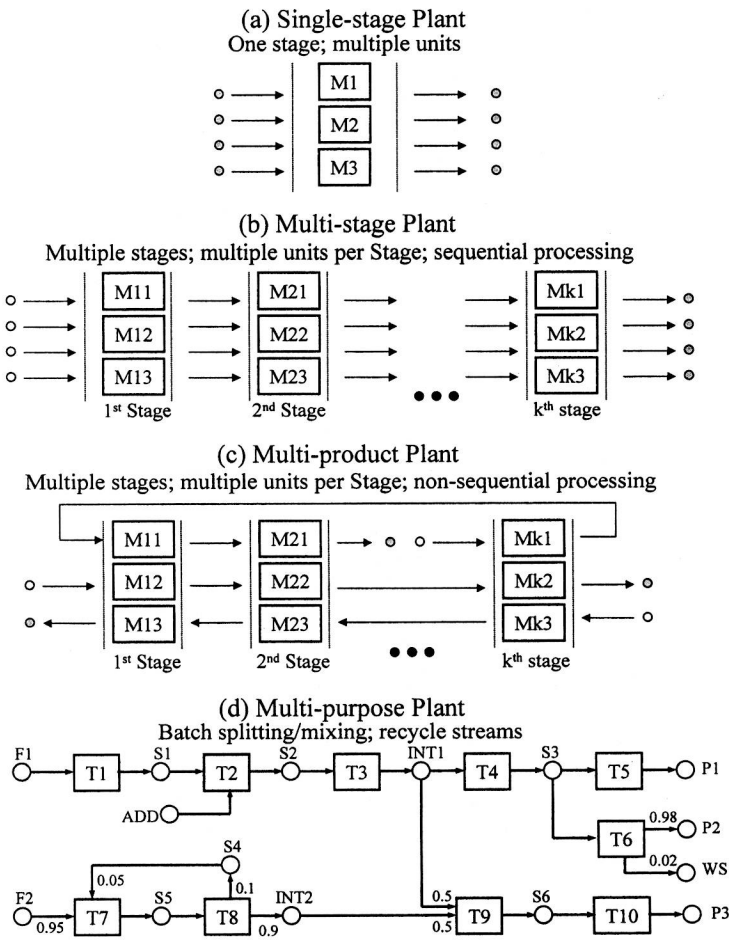


**Fig. 1.** Common plant configurations in chemical industry

Due to the different plant configurations and process specifications, a wide variety of optimization models, mainly MILP models, have been proposed in the process systems engineering literature. In multipurpose batch plants, for instance, where the level of inventories and the level of resource consumption should be monitored and constrained, the time horizon is partitioned into a sufficiently large number of periods, and variables and constraints are defined for each time period. In single-stage plants, on the other hand, where no batch splitting, mixing and recycle streams are allowed, batch sizes are usually assumed to be constant and this, in turn means that mass balance equations need not be included in the formulation. If, in addition, there are no utility constraints the time horizon is not partitioned into common time periods, and hence, assignment binaries are indexed by tasks and unit time slots, which are generally fewer than time periods. When the number of tasks and the assignments of tasks to units are known, assignments binaries and constraints are dropped, and sequencing binaries are used instead.

In an effort to develop a general representation that can be used in the majority of scheduling problems of chemical industry, Kondili et al. [8] proposed the discrete-time State Task Network (STN) representation, and the equivalent Resource Task Network (RTN) representation was proposed by Pantelides [9]. To overcome some limitations of discrete-time models, several authors proposed continuous-time STN/RTN models ([10], [11], [12], [13], [14]). While being very general, STN and RTN-based models are computationally inefficient for many classes of problems. The computational performance of both models is very poor, for instance, when the objective is the minimization of makespan for a given demand. Finally, STN-based models do not exploit the special structure of simple configurations, being orders of magnitude slower than special purpose MILP models.

To address these issues, Maravelias and Grossmann proposed a general hybrid MILP/CP iterative algorithm ([1], [2]) for the scheduling of multipurpose plants that exploits the complementary strengths of Mathematical and Constraint Programming. In this paper we show how this hybrid algorithm can be modified to address scheduling problems in multi-stage and single-stage plants. It is shown that the same idea can be used in all these configurations: use MILP to find a partial solution that includes the type and number of tasks and the assignments of units to tasks, and use CP to check feasibility, generate integer cuts and derive complete schedules.

In the section 2 we briefly present the hybrid algorithm of Maravelias and Grossmann. In section 3 we present the different types of problems, and in sections 4 and 5 we present how this algorithm can be modified to address these problems. Finally, we report computational results to show that order-magnitude reductions in computation time are possible with the proposed hybrid schemes.

## 2 Hybrid MILP/CP Algorithm

The main idea of the proposed algorithm is to use MILP to optimize (identify partial, potentially good solutions), and CP to check feasibility and derive complete, feasible schedules. Specifically, an iterative scheme where we iterate between a MILP master problem and a CP subproblem is proposed. The type and number of tasks to be performed and the assignment of tasks to equipment units are determined in the