APPLIED OPTIMIZATION

Alexandre Dolgui
Jerzy Soldek
Oleg Zaikin

SUPPLY CHAIN OPTIMISATION

Product/Process Design, Facility Location and Flow Control

Springer

SUPPLY CHAIN OPTIMISATION

Applied Optimization

Volume 94

Series Editors:

Panos M. Pardalos
University of Florida, U.S.A.

Donald W. Hearn
University of Florida, U.S.A.

SUPPLY CHAIN OPTIMISATION

Product/Process Design, Facility Location and Flow Control

Edited by

ALEXANDRE DOLGUI
Ecole des Mines de Saint Etienne, France

JERZY SOLDEK
Technical University of Szczecin, Poland

OLEG ZAIKIN
Technical University of Szczecin, Poland

Springer

eBook ISBN: 0-387-23581-7
Print ISBN: 0-387-23566-3

Created in the United States of America

Contents

This page intentionally left blank

Contributing authors

Aguirre, Omar
Panamerican University, Mexico

Aldanondo, Michel
Ecole des Mines d'Albi Carmaux, France

Amari, Said
Ecole des Mines de Nantes, France

Banaszak, Zbigniew
University of Zielona Gora, Poland

Bargelis, Algirdas
Kaunas University of Technology, Lithuania

Beaune, Philippe
Ecole Nationale Supérieure des Mines de Saint Etienne, France

Benyoucef, Lyès
INRIA - Lorraine, Metz, France

Bostel, Nathalie
IUT de Saint-Nazaire, France

Boutevin, Corinne
Université Blaise-Pascal, Clermont-Ferrand, France

Campagne, Jean-Pierre
Institut National des Sciences Appliquées de Lyon, France

Chauhan, Singh Satyaveer
INRIA-Lorraine, Metz, France

Dejax, Pierre
Ecole des Mines de Nantes, France

Demongodin, Isabel
Ecole des Mines de Nantes, France

Deroussi, Laurent
Université Blaise-Pascal, Clermont-Ferrand, France

Ding, Hongwei
INRIA – Lorraine, Metz, France

Dolgui, Alexandre
Ecole Nationale Supérieure des Mines de Saint Etienne, France

Eremeev, Anton
Omsk Branch of Sobolev Institute of Mathematics, Russia

Fleming, Peter
University of Sheffield, UK

Galland, Stephane
Université de Technologie de Belfort-Montbéliard, France

Geneste, Laurent
Ecole Nationale d'Ingénieurs de Tarbes, France

Genin, Patrick
Ecole Nationale Supérieure des Mines de Paris, France

Gourgand, Michel
Université Blaise-Pascal, Clermont-Ferrand, France

Grabis, Janis
Riga Technical University, Latvia

Grabot, Bernard
Ecole Nationale d'Ingénieurs de Tarbes, France

Grimaud, Frédéric
Ecole Nationale Supérieure des Mines de Saint Etienne, France

Hadj-Hamou, Khaled
Ecole des Mines d'Albi Carmaux, France

Haustein, Jorg
OR Soft Jänicke GmbH, Germany

Henoch, Bengt
Jönköping University, Sweden

Janicke, Winfried
OR Soft Jänicke GmbH, Merseburg, Germany

Kazheunikau, Mikhail
Belarusian State University of Informatics and Radioelectronics, Minsk, Belarus

Kolokolov, Alexander
Omsk Branch of Sobolev Institute of Mathematics, Russia

Korytkowski, Przemysław
Technical University of Szczecin, Poland

Kosanke, Kurt
CIMOSA Association e.V., Germany

Kulikov, Gennady
Ufa State Aviation Technical University, Russia

Kushtina, Emma
Technical University of Szczecin, Poland

Lamothe, Jacques
Ecole des Mines d'Albi Carmaux, France

Lamouri, Samir
Ecole Nationale Supérieure des Mines de Paris, France

Loiseau, Jean-Jacques
Ecole Centrale de Nantes, France

Lu, Zhiqiang
Ecole des Mines de Nantes, France

Małachowski, Bartłomiej
Technical University of Szczecin, Poland

Mankutė, Rasa
Kaunas University of Technology, Lithuania

Merkuryev, Yuri
Riga Technical University, Latvia

Merkuryeva, Galina
Riga Technical University, Latvia

Norre, Sylvie
Université Blaise-Pascal, Clermont-Ferrand, France

Pashkevich, Anatoly
Belarusian State University of Informatics and Radioelectronics, Minsk, Belarus

Pashkevich, Maxim
Belarussian State University, Minsk, Belarus

Petuhova, Julija
Riga Technical University, Latvia

Polak, Michał
Advanced Digital Broadcast Polska Ltd., Poland

Sandkuhl, Kurt
Jönköping University, Sweden

Servakh, Vladimir
Omsk Branch of Sobolev Institute of Mathematics, Russia

Smirnov, Alexander
St. Petersburg Institute for Informatics and Automation, Russia

Sotskov, Yuri
Institute of Engineering Cybernetics, Minsk, Belarus

Sotskova, Nadezhda
Hochschule Magdeburg-Stendal, Germany

Thamelt, Wolfgang
OR Soft Jänicke GmbH, Merseburg, Germany

Thomas, André
Ecole Nationale Supérieure des Technologies et Industries du Bois,
Epinal, France

Werner, Frank
Otto-von-Guericke-University, Magdeburg, Germany

Worley, Jorge Hermosillo
Ecole Nationale d’Ingénieurs de Tarbes, France

Xie, Xiaolan
INRIA-Lorraine, Metz, France

Zaikin, Oleg
Technical University of Szczecin, Poland

This page intentionally left blank

Preface

The idea of this book came to us in October 2002 at the international conference "Production systems design, supply chain management and logistics" (Miedzyzdroje, Poland). This conference was organized by an initiative group from certain French and Polish Universities with the objective to find a new synergy and to develop a Pan European (West – East) cooperation in the areas of:

Design of production systems
Supply and inventory management
Production planning and scheduling
Facility location, transportation and logistics
Supply chain optimisation via simulation

The conference was focused on a wide spectrum of optimisation problems taking into account Supply Chain paradigms, which create a pivotal idea to increase the productivity of modern production systems.

The editors proposed to the group members, to present their results for this book. Some well-known scientists agreed also to participate in this work and sent us their contribution.

The main idea of this book is that now, it is necessary to consider all the stages of product life cycle in an integrated approach, from the product/process design to the customer delivering. For example at the product design stage, we need to analyse and to optimise all the supply chain.

The book is composed of 20 chapters divided into three parts.

The first part of the book presents a set of modelling techniques taking into account the enterprise integration problem (K. Kosanke), the knowledge management in the SME networks (K. Sandkuhl *et al.*), and the human resources in business process engineering (J. Worley *et al.*). It deals with advanced demand forecasting methods (J. Petuhova and Y. Merkuriev;

M. Pashkevich and A. Dolgui), deadlock avoiding in utilisation of common resources (Z. Banaszak and M. Polak), sizing and plant control using dioid algebra (S. Amari *et al.*) and queueing modelling of resources allocation for distributed e-production (O. Zaikin *et al.*).

The second part is dedicated to advanced optimisation methods. The assembly line balancing problem (load balancing) is explored in two ways: mathematical analysis of stability of optimal solutions (Yu. Sotskov *et al.*) and search for an optimal solution using meta-heuristics techniques (C. Boutevin *et al.*). New analytical approaches to facility allocation (Z. Lu *et al.*), supply chain design (J. Lamothe *et al.*), sales and operations planning (P. Genin *et al.*) and delivery cost optimisation (S. Chauhan *et al.*) are also presented. These are completed by an interesting optimisation technique which is based on simulation and response surface methods (G. Merkurieva).

In this domain, the research activities are directly linked to real industrial problems. Therefore, it is necessary to develop applied decision aid tools. The third part of this book deals with some examples of these tools. A special tool for supply chain simulation (H. Ding *et al.*), a Web-based tool for product/ process integration (A. Bargelis and R. Mankute), a chemical plant scheduling tool (N. Sotskova *et al.*), maintenance planning optimisation tools (Pashkevich et *al.*) and a multi-agent tool and a software platform architecture for distributed industrial systems analysis (S. Galland *et al.*) are presented.

We acknowledge all the reviewers and the authors for their contribution to this book. We hope that this book will be useful to the whole community of scientists in Computer Science and Industrial Engineering.

Saint Etienne, France
Prof. Alexandre Dolgui

Szczecin, Poland
Prof. Jerzy Soldek
Prof. Oleg Zaikin

PART I: MODELLING TECHNIQUES

This page intentionally left blank

Chapter 1

AN INITIATIVE FOR INTERNATIONAL CONSENSUS ON ENTERPRISE INTER- AND INTRA-ORGANISATIONAL INTEGRATION

Kurt Kosanke

Abstract: The initiative is aimed on building international consensus on enterprise integration and is carried out in five-year intervals. The third initiative focused on virtual enterprises and in particular on aspects of enterprise engineering, relations between knowledge management and business process modelling, the issue of interoperability of business processes and models and the needs of common model representation. The roles of ontology and agent technologies have been addressed as potential solutions for the current issues in enterprise integration. Four workshops with international experts discussed the different issues and developed proposals for solutions of the issues identified. Workshop results were presented at the International Conference on Enterprise Integration and Modelling Technologies (ICEIMT'02) and are published in the proceedings of the initiative. The paper summarises the results of the initiative including some details from the four workshops and provides an outlook on future activities resulting from the initiative.

Key words: business process modelling, enterprise integration, enterprise interoperability, enterprise modelling, knowledge management.

1. INTRODUCTION

Virtual Enterprises (VE) and Business to Business (B2B) type applications of electronic commerce are new ways - especially for Small and Medium Enterprises (SME) - to unite forces, increase their competitiveness, meet today's market needs and jointly behave as one producer towards the customer. However, the main concern in the required collaborations is the need for trust between the partners, which can certainly be enhanced by sufficient information on partner capabilities. Those capabilities can best be described through models of the relevant business processes and their associated information and resources. Linking compatible business process models from the different collaborators into an overall business process model will allow evaluating the collaboration prior to its real implementation through a priori simulation of the intended operation. Up to now application of relevant Information and Communication Technology support has been hampered by a lack of business justification, by a plethora of seemingly conflicting solutions and confusing terminology [2,6], and by an insufficient understanding of the technology by the end-user community.

The third international initiative had again the objective to increase both international consensus and public awareness on enterprise integration [1,3]. Following the two previous initiatives in 1992 and 1997 [7,4], the focus of the third initiative was on Enterprise Inter- and Intra-Organisational Integration. Drivers, barriers and enablers for electronic commerce in general and B2B applications in particular, as well as potential benefits from the application of integration supporting information and communication technology have been addressed.

Up to 25 selected experts in the fields of engineering, business administration, and computer science participated in each of the four workshops. About 75 persons from 18 countries on 5 continents attended the ICEIMT’02, coming from academic institutions, government, industry, and consortia. The conference proceedings [3] provide about 40 papers offering a very comprehensive overview of the state-of-the-art in of enterprise integration as well as providing directions for further research in the 9 working group reports, which present the different workshop results.

2. METHODOLOGY AND ACTIVITIES

The international initiative on Enterprise Inter- and Intra-Organisational Integration (EI3-IC) has provided a basis for an international discourse on the subject of enterprise inter- and intra-organisational co-operation. Inviting experts in the field has enabled pulling in insights and results from other projects hence enabling a consolidation of this fragmented know-how, and thereby contributing to an international consensus in the field. Therefore, it enables the presentation of both current status and potential developments in inter- and intra-organisational integration for electronic commerce with focus on B2B applications.

The EI3-IC initiative consists of two parts:

1. Four workshops with international experts reviewing and consolidating a set of issues in enterprise inter and intra-organisational integration.

2. The ICEIMT'02 (International Conference on Enterprise Integration and Modelling Technologies) aimed on state of the art overview and presentation of the workshop results.

A scientific committee guided and supported the initiative. It acted as advisory committee helping to identify the experts to be invited to the workshops and reviewing workshop and conference agendas and papers.

The 3-day workshops have been organised with plenary sessions for all participants and a number of parallel working group sessions. The first plenary session held in all workshops provided time for the participants to present their own work as it relates to the predefined set of issues. This methodology has led again to very good results. It enables the members of the working group to have a common understanding of each other's position leading to a much better focusing on the issues to be discussed.

During the first plenary session the experts will usually amend the set of predefined issues. Working groups have then worked on subsets of the issues of the particular workshop. Presentation of working group results and discussions of the topic with all working groups have been done during subsequent plenary sessions.

Papers on workshop results were prepared co-operatively by the working groups and presented at the ICEIMT'02 by a group member. Information dissemination activities will further increase awareness and consensus within academia and industry about enterprise inter-and intra-organisational integration.

2.1 Workshops and Conference

Four thematic workshops with international experts in the field have been organised. The workshop themes have been selected according to their importance for the management of business collaborations. The following workshops have been held:

- Workshop 1, Knowledge management in inter- and intra-organisation environments (EADS, Paris, France, 2001-12-05/06)
- Workshop 2, Enterprise inter- and intra-organisation engineering and integration (Gintic, Singapore, 2002-01-23/25)
- Workshop 3, Interoperability of business processes and enterprise models (NIST, Gaithersburg, MD, USA, 2002-02-06/08)
- Workshop 4, Common representation of enterprise models (IPK, Berlin, Germany, 2002-02-20/22)

The ICEIMT'02 was held at the Polytechnic University of Valencia, Spain, on 2002-04 24/26. It was structured according to the themes of the workshops. In addition to an opening session with keynote papers, a special session on international projects provided information on actual work done on an international level.

3. RESULTS

As stated in [5] the results from all four workshops indicate the important role of business processes in the area of e-commerce and virtual enterprises. Sharing relevant knowledge between co-operating partners and making it available for decision support at all levels of management and across organisational boundaries will significantly enhance the trust between the partners on the different levels of partner operations (strategy, policy, operation and transaction). Clearly business process modelling can significantly enhance establishment, operation and decommission of the required collaboration. However, interoperability between partner processes and common understanding of their model semantics is a prerequisite for successful collaborations.

Agent technology has been a subject in all four workshops as well and several proposals for further work have been made. The same is true for the concept of ontology, which will play an important role in solving the interoperability issues and provide for common understanding through the harmonisation of business knowledge semantics.

More specific results from the four workshops are presented in the following sections. Tables 1-1 to 1-4 (derived from the different working group reports in [3]) identify major issues discussed and results obtained from the working groups.

3.1 Knowledge Management and Business Process Modelling in Inter- and Intra-Organisational Environments (Workshop 1)

Knowledge management has gained significant momentum within enterprise organisations and is considered an important success factor in its operation. However, there exist wide differences in the understanding of what a knowledge management system is and does. The perception of knowledge management ranges from using enterprise-wide databases or expert systems to enterprise modelling and integrated communication systems, which are to be supported by Internet technology. Generally accepted guidelines or standards to support the design and implementation of a knowledge management system in an organisation or between organisations are missing. Capturing knowledge and using it across organisational boundaries with a satisfactory acceptance of the human user is another major challenge.

Table 1-1. Issues and Results from Workshop 1

Major problems and issues:	**Results and *future work* needed:**
Working Group 1	
– Awareness and education at all organisational levels – KM metrics for cost-benefit analyses – Auditability of intellectual property – Softness of many KM topics – Lack of enterprise-wide continuity in KM systems – Diverse corporate culture in virtual and merged enterprises	– Knowledge exists only in human minds-stuff stored electronically is information – No new techniques are needed to model information relating to knowledge – *Methods for representing information about "soft" enterprise activities (strategic planning and decision making)* – *Methodology to define what we know, need to know, do not know, cannot know and what to forget at what time.*
Working Group 2	
– Creation and exploitation of synergy between KM and BPM – Integration of general knowledge into business process models – Identification of critical knowledge in business processes? – Role of ontologies in KM and BPM?	– KM and BPM have common objectives (capture, structure and provide knowledge for decision making) – Proposal for mapping KM onto BPM – *Establish a formal base for enterprise ontologies* – *Analyse the potential contributions of semantic web technologies*
Working Group 3	
– Lack of a common understanding and barriers for KM in industry – Scope and goal of KM to enable growth with the (system) life cycle and adaptation to evolving infrastructures – Use of existing standards – Guidelines for implementation and use of KM systems especially in SMEs	– Definition of Requirements for KM system infrastructures – Synthesis from examples of Process and KM applications – *Methodologies for scalable KM systems* – *Investigation of dependencies and interoperation of (process) model management and KM* – *Development of an infrastructure consisting of IT and non-IT services to support KM across organisational borders*

Merging Knowledge Management (KM) and Business Process Modelling (BPM) will provide synergy and improve efficiency of enterprise collaborations. During the workshop, three working groups addressed the relations between knowledge management and business process modelling concluding that joining both in some form could be possible and synergy would bring additional benefits (see also Table 1-1). The focus of the first working group was on possible combined futures and the research roadmap these futures require. Three different levels of potential work have been identified: near term, medium term and longer term oriented. Problems and limits at each level have been identified and potential solutions are proposed.

Discussing the mapping of BPM concepts onto KM concepts similarities and differences as well as solutions have been identified by the second working group. Ontology will play an important role in this mapping. This will become intensified even more with the move towards inter-organisational collaboration. The working group has started to map the two concepts into a common methodology.

Concentrating on guidelines for business process modelling to cover scope and goals, architectures, infrastructures and approaches to implementation, the third working group looked at examples of industrial solutions and tool strategies. Potential synergies and solutions have been identified with emphasis on the human role in future environments.

However, the benefits of knowledge sharing between collaborators can only be exploited if interoperability of business processes and business-process models can be assured. This is especially important during the virtual enterprise establishment phase where the required and provided capabilities have to be matched under the time constraints of the market window.

3.2 Enterprise Inter- and Intra-Organisational Engineering and Integration (Workshop 2)

Collaboration is not only a technical issue, but also a social and organisational one, as well as a matter of trust. That means enterprise engineering has to cover both of these aspects equally well. But there is a significant difference for the two subjects in the degree of understanding of the problems and of the potential solutions. Whereas technology behaviour is to a large extend predictable and technical issues are usually understood and mostly appear solvable, human behaviour is non-deterministic and solving human related so-called soft issues requires different methodologies.

This workshop addressed both of these topics focussing on infrastructures and on planning of virtual enterprises. The first working group proposes the exploitation of agent technology to obtain solutions applicable for advanced virtual enterprises. Such concept includes the use of agent-model pairs applying ontology and thereby addressing the issue of model semantics and its impact on model complexity and costs.

The second working group identified a set of common VE business planning activities and the degree of concurrency between planning processes at different planning levels. The concept of team building and the related human issues has been a special topic recognised in the working group discussions and in the proposed planning activities.

Table 1-2. Issues and Results from Workshop 2

Major problems and issues:	Results and *future work* needed:
Working Group 1	
– Employ new technologies (agents, models and there combinations) in inter-organisational enterprises – Determine if special modelling techniques are required to support enterprises driven by agents, actors and their models	– Agents using enterprise models are the triggers that enable model-driven enterprises to work – *Extend Process Specification Language to be more agent friendly and to include state mechanics* – *Develop index systems for existing self-organising model frameworks*
Working Group 2	
– Increase efficiency in the collaborations in virtual environments? – Define languages and methods to describe business strategies and business models in relations to the life cycle phases of the GERAM modelling framework	– Identification of a set of common VE business planning activities – Identification of degree of concurrency between planning processes at different planning levels – *Test concept in real practical applications* – *Investigate concept relations to concepts in human and management science* – *Investigate communication and negotiation needs with emphasis on human relations*

3.3 Interoperability of Business Processes and Enterprise Models (Workshop 3)

Integration is the timely and meaningful exchange of information among software applications. It requires the error-free transfer of information, a total agreement on its syntax, and the correct understanding of its semantics. The Internet and its associated standards have addressed successfully the first of these requirements. Syntax and semantics, on the other hand, remain as elusive today as they were ten years ago. These issues are resolved typically by proprietary, de facto, or standard-interface specifications, which, in theory, should have solved the problem, but have not because the costs of development and custom implementation remain prohibitively high.

Two working groups addressed the issues of systems requirements and the role of ontology from a business process integration point of view (see also Table 1-3). Discussions in the first group were on life-cycle-based system engineering and how to interoperate across the different engineering life-cycle phases and between their different processes in the enterprise. Emphasis was on product development and production processes development.

The second group addressed the barriers of enterprise integration and examined the new leverage that ontology might provide. The group agreed that such an approach could overcome the most severe of these barriers - the lacking common semantics. A number of actions and proposals have been outlined by the group, which may be taken up especially in NIST (National Institute of Standards and Technology) activities. But interoperability has not only an information technology aspect, but a human aspect as well. Only if the business-process model representation is commonly understood, will the people involved in the collaboration be able to build and maintain the needed trust in each other's capabilities.

Table 1-3. Issues and Results from Workshop 3

Major problems and issues:	**Results and *future work* needed:**
Working Group 1	
– Interoperability of processes and models – Model of interactions between all life cycle activities of both production- and product processes – Concurrent use of product design and production system engineering data – Synthesis of data dictionaries	– Interoperability: on-time transfer of understood information between processes. – Metrics for interoperation quality: number of conversations needed--not needed – *Define relevant processes to support design optimisation and production decision.* – *Emphasise human-oriented information exchange* – *Define set of required standards*
Working Group 2	
– High and unpredictable cost of enterprise modelling. – Models are often too complex to use. – How to use ontology for more efficient modelling	– Model complexity is due to the semantic content that overloads models. – *Evaluate ontologies for their ability to:* – *Improve cost versus benefits by adding formal rigor to ontologies* – *Separate semantic aspects of models from non-semantic ones* – *Improve agent's use of knowledge* – *Improve security in information sharing by assigning a context code to each shared parcel of information.*

3.4 Common Representation of Enterprise Models (Workshop 4)

Many industrial users think of models as a blueprint of the enterprise. As this has been the case originally, it is not true any more. Enterprise models or business-process models nowadays not only provide an understanding about the enterprise operation, but also are actively used for knowledge management, decision support through simulation of operation alternatives and even for model-based operation control and monitoring.

Whereas before model creation was a skill left to experts, it will become a need for many people in the enterprise to be able to evaluate process alternatives for decision support. That means we need executable models as well as a common representation of the models for the model users to enable understanding and easy manipulation of the models. However, common representation does not imply an Esperanto like language, but rather a set of dialects aimed at the different user groups, but based on a common set of modelling language constructs.

Table 1-4. Issues and Results from Workshop 4

Major problems and issues:	**Results and *future work* needed:**
Working Group 1	
– Convince users of the value of EM – Gap between user expectations towards EM and modelling expert results – Faithfulness of models to the reality and the maintainability of models – EM to enable model based decision support – Guide the user in modelling and evaluating process alternatives? – Link business process models to the actual operational data bases of the enterprise	– Business process model to be the blueprint of the enterprise – Outline of user enabled business process modelling directed towards model based decision support – *Identify the common set of modelling language constructs (e.g. UEML) from which the representations needed by the different users can be derived* – *Develop the methodologies to support the users in modelling and evaluating business alternatives*
Working Group 2	
– EM encompasses local and global views with different terminology, modelling methods, and ontologies; and occupies different space in virtual enterprises – Globally modelled things tend to be "soft" and non-deterministic – The infinity of tacit knowledge needs to be classifiable into apropos chunks	– *Develop methodologies to support:* – *Soft modelling: uncertainty, non-determinism, social and cultural dynamics, and tacit knowledge* – *Introspective modelling: models to control other models* – *Multi-world modelling: legal, financial, and production domains contain conceptual discontinuities.* – *Multi-level modelling: interaction between process models and enterprise models* – *Meta-tools for modelling to support multi-level and multi-world models*

The two working groups addressed the user orientation and new support technologies for enterprise integration (see also Table 1-4). Emphasis has been placed in working group 1 on the need for user oriented business process modelling, which is seen as a prerequisite for model based decision support. Critical issues discussed include the role of the user and his requirements in the modelling process. Especially emphasis was on the use of current process information needed to evaluate proposed solutions and the use of formal methods for semantic mappings between different tools and models. The working group explored methodologies needed to support user-enabled business process modelling for model based decision support.

The second working group focused on radical but practical strategies for greatly improving process modelling in an enterprise context. The group's work centred on improving user benefits in the context of common models, enterprise context and enterprise views. Major problems addressed were: multi-world views, soft modelling and meta-modelling theories. Several discrete research projects are proposed.

4. CONCLUSIONS

Considering the results from the workshops as identified in the different working group reports several conclusions can be drawn:

- On Knowledge Management: merging knowledge management and business process modelling will significantly improve decision support at all levels of the enterprise by providing more relevant knowledge structured according to the business processes of the enterprise.
- On Enterprise Engineering and Interoperability: the current issue of process semantics should be addressed by employing ontologies both for improving both human and machine understandability and inter-organisational interoperability.
- On Model Representation: focus has to be on the industrial user and his need for model based decision support.

All working group identified the need for further enhancements of the state of the art by identifying significant subject for further research and development. Some of the efforts identified will be taken up by research projects, other still need the critical mass to be assembled.

International consensus on the contents of enterprise intra- and inter-organisation integration is a prerequisite for real industry acceptance and application of the technology. With its particular focus on e-commerce the third initiative identified major players in this field both in industry and academia and thereby has continued to build the community on enterprise integration. A community that will continue the drive for consensus beyond this initiative and towards a follow-on ICEIMT and will continue as an international forum to further identify and eliminate barriers to utilisation of inter- and intra-organisational integration technology [5].

The next international conference on Enterprise Integration and Modelling Technology (ICEIMT) will breakout of the five year interval of the previous conferences and will already be held on 2004-10-09/11 at the University of Toronto. The conference will again focus on the subject of inter and intra-organisational integration, with emphasis on enterprise knowledge, modelling languages and ontologies and the potential support by the semantic web as well as supporting infrastructures. International standards and common terminology are further subjects to be addressed in the next ICEIMT.

However, significant efforts are still needed to gain awareness and acceptance in the industry. Large-scale demonstrations of pilot applications as well as professional training of potential users would be means to convince the user community of the benefits of the technology. Standardisation as well has to play an important role in this international consensus building and awareness and acceptance tasks. Only with relevant standards can the inter-organisational collaboration in the e-business environment be achieved that the new organisation paradigms like virtual enterprise predict. Such standards are needed in the area of supporting tools and languages as well as in the operational infrastructures for knowledge sharing and information exchange.

ACKNOWLEDGMENTS

The European Commission, DG Information Society, under IST 2001-92039 and the International Federation of Information Processing - IFIP TC5/WG 5.12, supported the initiative. Initiative organisation was done jointly by the CIMOSA Association, Germany, the Fraunhofer Institute IPK (Institute for Production Systems and Design Technology), Germany, the National Institute of Standards and Technology (NIST), USA and the Polytechnic University of Valencia, Spain.

The author likes to thank all the organisations involved in this initiative for their support and their contributions without which this initiative would not have taken place.

Special thanks to the organisations hosting the different workshops – EADS, Paris, France; Gintic, Singapore, Rep: of Singapore; NIST, Gaithersburg, USA; IPK, Berlin, Germany – and to the Polytechnic University of Valencia, Spain for hosting the ICEIMT'02.

Not forgotten are the efforts of the scientific and the technical committee and the reviewers. Thank you for the many very valuable suggestions during the course of the initiative as well as in the review process.

Last but not the least my sincere thanks to all workshop and conference participants. It's about your work that is reported in this paper. The many presentations and discussions that have been made by you indicate your dedication and your strong commitment to the subject of enterprise integration.

REFERENCES

1. Goranson, H. T, 2002 ICEIMT: History and Challenges. In [3] below.
2. Kosanke, K. de Meer, J. 2001 Consistent Terminology - A Problem in Standardisation – State of Art Report of Enterprise Engineering. *Proceedings SITT'01,* Boulder, Col, USA, October 3/5.
3. Kosanke, K. Jochem, R. Nell, J.G. Ortiz Bas, A. (Eds.) 2002 Enterprise Engineering and Integration: Building International Consensus. *Proceedings of ICEIMT'02 Intern. Conference on Enterprise Integration and Modelling Technology;* Kluwer Academic Publisher
4. Kosanke, K. Nell, J.G. (Eds.), 1997 Enterprise Engineering and Integration: Building International Consensus. *Proceedings of ICEIMT'97 Intern. Conference on Enterprise Integration and Modelling Technology;* Springer-Verlag.
5. Nell, J.G. Goranson, H. T. 2002 Accomplishments of the ICEIMT'02 Activities. *In* [3] above.
6. Söderström, E., 2002 Standardising the business vocabulary of standards *Proceedings ACM Symposium on Applied Computing (SAC 2002),* March 10-14, Madrid, Spain.
7. Petrie, Jr. C.J. (Ed.), 1992 Enterprise Integration Modelling. *Proceedings of the First International Conference,* MIT Press.

Chapter 2

TOWARDS KNOWLEDGE LOGISTICS IN AGILE SME NETWORKS

A Technological and Organisational Concept

Kurt Sandkuhl, Alexander Smirnov, Bengt Henoch

Abstract: Due to globalization and increased competition, the future market position of small and medium-sized enterprises (SME) is closely related to the ability of cooperating with partners and of reusing existing knowledge. Solutions for efficient knowledge logistics will form a key success factor for distributed and networked enterprises. In our approach, we consider competence models as a knowledge source in SME networks and we use knowledge supply networks as an infrastructure for knowledge logistics. The chapter introduces organizational and technological aspects of our approach including university – SME cooperation, semantic nets and multi-agent framework. Application area is the field of agile SME-networks which are typically temporary, dynamical with respect to their members, geographically distributed, and flexible to market demands.

Key words: knowledge supply net, competence model, agile SME-network, semantic net.

1. BACKGROUND

Due to globalization and increased competition, the future market position and competitiveness of small and medium-sized enterprises (SME) is closely related to the ability of cooperating with partners. In several industry fields (e.g. automotive, aerospace, print & media) this is reflected in a trend to virtual supplier organizations loosely integrating enterprises based on their contribution to the value chain [7]. Other examples for SME-networks include temporary project-oriented co-operations (e.g. in product development or construction projects), trade organizations, and associations for joint marketing activities.

In our research work, we are especially interested in agile SME-networks. Agile SME networks are communities or associations of SMEs based on common economical and value-creation objectives. They pro-actively form co-operations for joint product development or project work. These co-operations typically are temporary, dynamical with respect to their members, geographically distributed, flexible and quick responsive to market demands.

Due to geographic distribution and dynamics with respect to network members, optimized knowledge supply and efficient re-use of existing knowledge is a critical success factor for agile SME-networks. Usually it is not fully transparent to the network members, which knowledge is available in which intensity at which partners site to which costs and how to access it. One of the major constraints to the success of SME-networks is the difficulty of collectively bringing together many disparate enterprises, consultants, and other participants, and ensuring a common level of knowledge, understanding, and commitment. SME-networks require cooperation and an open exchange of information among all participants.

As a consequence, we propose to implement a cost-effective Knowledge Source Network (KSNet) for intra-network knowledge exchange and supply. In our approach, we consider competence models of SMEs, electronically stored information and personnel resources in enterprises as knowledge sources, which are integrated into a joint infrastructure based on a KSNet.

Section 2 introduces organizational and technological concepts for competence modeling of our approach. Section 3 afterwards investigates concepts of KSNet. As competence modeling has been used in the ”SME-Chains” project, first experiences and conclusions are presented in Section 4.

2. COMPETENCE MODELLING

2.1 Organizational Aspects

Identification of potential sources for knowledge in SMEs results in three main categories:

(1) Most of the knowledge exists as *competences of employees,* who very often exercise several roles in the enterprise simultaneously. Personal skill profiles can serve as a description of this knowledge. This field has been investigated in several research projects, e.g. [3, 10] and is not subject of this work.

(2) *Externalized knowledge* stored electronically in documents, databases or information systems. These knowledge objects can be office documents

(e.g. design rules from manufacturer for product development at supplier), CAD drawings of parts or sub-parts, executable routines for simulation of processes or machinery, or formal requirement specifications from the customer.

(3) *Corporate knowledge* represented in work processes, organizational structures, standard operation procedures, or best practices. In most SMEs, this knowledge has not been documented and externalized.

In this section we focus on the latter knowledge category. We consider competence models in SMEs as promising way to capture this knowledge and will introduce organizational aspects of our approach. To our understanding, competence information ideally has to encompass all technical and organizational capabilities of an enterprise. This includes

– Skill profiles of the personnel of the enterprise,
– Technical equipment and production capacity,
– Business processes with focus on value creation and management processes,
– Organizational capabilities,
– Technical and service-oriented products with their features and parameters.

Competence modeling therefore is closely related to enterprise modeling [11]. Competence modeling is a non-trivial task requiring solid competences in information modeling which cannot be taken for granted in SMEs. Organizational support for competence modeling of SMEs therefore is of crucial importance and has to take into account regional and cultural aspects and the individual needs of SMEs. Our concept for organizational support of SMEs is to integrate universities as partner organizations into regional SME-networks and to use student ambassadors as contact partners and assistants to SMEs.

On the regional level each SME participating in the SME-network will have close contacts with the students and receive visits from a team of two students from the regional university. This visiting student team assists in modeling the enterprises competences and generating the meta-data in accordance to the semantic net (see Section 2.2), integrating the competence model as a knowledge source into the KSNet. An SME can also be host for a pair of students during their study time. These students will work on an individual "mini-project" for the host SME, which will be integrated into his or her studies at the university. The regional university, which from the SMEs viewpoint is considered as competent and trustworthy partner, supports the students by offering courses in relevant engineering methods and in project management.

This type of university – SME cooperation generates benefits for all partners: The students benefit by gathering practical experience from their project. The SMEs get individual support in their projects and – as a side effect – close contacts to well educated potential future employees. The university has benefits by learning the everyday needs of SMEs in the region.

2.2 Technological Aspects

Knowledge supply has to be more than simply installing a search engine or providing a joint repository for all partners. Our approach is to use semantic modeling in order to improve knowledge capturing, knowledge reuse and knowledge transfer. Semantic modeling means in this context that the semantics of all relevant concepts of the industry sector or domain will be modeled in a semantic net by capturing the associations between the concepts. This semantic net, which is a pre-stage for an ontology, defines a "common understanding" for the application area and helps to identify and find the most relevant information for a user by reflecting the meaning of the concepts.

The most important relationship amongst concepts is the so-called *subsumes* relationship. This expresses a hierarchical order and sums up the aggregation and generalization / specialization relationship. In addition, associations can be made, placing various concepts in a named relationship.

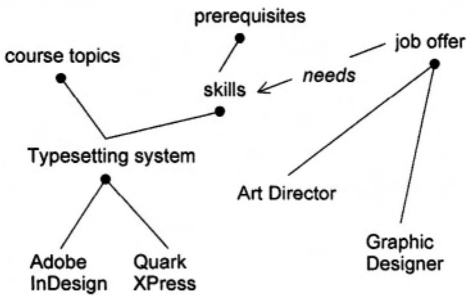


Figure 2-1. Extract of a semantic net from the domain "print & media"

Let us take the field of qualification and job offers in the area of print & media as an example. The depicted extract of the semantic net (see Figure 2-1) lists various concepts and their relationships[1]:

Semantic nets are not only used for stating classical relationships like „Adobe InDesign *is a* typesetting system" or „art director *is a* job offer" but also for stating the so-called *context* relationship, e.g. „Quark Xpress in context of typesetting".

Based on a semantic net especially developed for the agile SME-network in question, the existing knowledge is captured by using *concept paths.* For every knowledge object, a set of paths in the semantic net is identified. One path consists of a number of connected concepts in the semantic net, describing the characteristics of the knowledge object. These concept paths

[1] The edge with the filled-in circle depicts the "subsumes" relationship, while the arrow shows the named and specified association.

serve as meta-data in the KSNet when identifying and transferring the right knowledge with respect to a users' need. The concept paths and other meta-data of the knowledge object are stored in a *knowledge source repository.*

As modeling of all competence aspects for a large number of SMEs is time and cost intensive, we differentiate competence models and short competence profiles.

Competence models are prepared for an SME with the assistance of student ambassadors. The student uses an enterprise modeling tool (e.g. METIS toolset[2]) and templates prepared for the relevant industry sector. Modeling includes all aspects introduced in Section 2.1 and is done in close cooperation with the personnel of the enterprise. For the resulting competence model concept paths within the semantic net are generated serving as meta-data for knowledge selection and supply.

Competence profiles can easily be entered by the SMEs themselves via the Internet by using functionality of the knowledge source repository. The repository provides a profile entry form, which consists of properties, references to resources and concept paths in the semantic net. Properties are structured data types and used for representing meta-information on the enterprise, e.g. general contact information, available production capacities and personnel, etc. Resource reference provide the possibility to include additional material on the enterprise, e.g. text documents or product information. The most important part of the competence profiles is the identification of concept paths within the semantic net. These paths reflect the competences of the enterprise with respect to skills and product structure. The repository supports this task by providing a browser for semantic nets.

3. KSNET APPROACH

The term KSNet has its origin in the concept of virtual organization / enterprises [13, 16] based on the synergetic use of knowledge from multiple sources [8, 12, 14]. A KSNet can be defined as a flexible connection of appropriate knowledge sources at different locations with the target to fulfill a concrete task, e.g. the decision making of a task in a determined volume, cost frame and time. In the context of agile SME-network, the consortium exists for a period of time, with formation, integration and operation phase. The network becomes operational when a concrete realization takes place or at least the necessary budget is endorsed. During the planning phase until the offer is ratified the KSNet represents a planning task in order to design and evaluate potential scenario solutions for the decision making of task. Figure 2-2 explains roughly the basic concept of the KSNet.

[2] http://www.computas.no

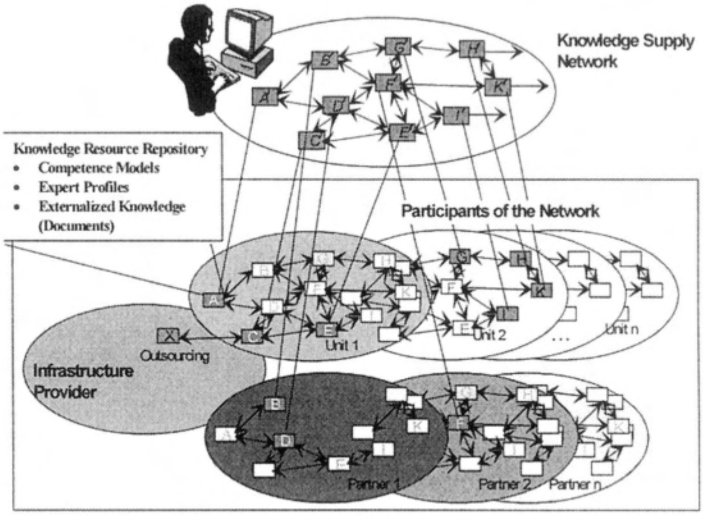


Figure 2-2. Distributed multi-level KSNet

In agile SME-networks, the knowledge sources interlinked and integrated by KSNet are expert profiles, externalized knowledge (electron. stored) or competence models (see Section 2). The semantic net supports navigation and knowledge supply.

Analysis of some existing systems/projects for knowledge source integration has shown several multi-agent approaches. Some examples are [1,9,12]:

- KRAFT (Knowledge Reuse and Fusion / Transformation) – multi-agent system for integration of heterogeneous information systems. The main aim of this project is to enable sharing and reuse of constraints embedded in heterogeneous databases and knowledge systems.
- InfoSleuth – multi-agent system for retrieving and processing information in a network of heterogeneous information sources.
- OBSERVER (Ontology Based System Enhanced with Relationship for Vocabulary heterogeneity Resolution) – system for information retrieving from repositories. The main aim is to retrieve information from heterogeneous knowledge sources without having knowledge of their structure, location and existence of the requested information.

An effective KSNet is characterized by (i) increased connectivity between its units, (ii) alignment of its inter-organization support systems, and (iii) sharing of information resources among its units.

Major KSNet functions could be determined as (i) communication, (ii) co-ordination, (iii) collaboration, and (iv) common/shared memory. This set of functions could be realized by using the following technologies [14]:

- Direct knowledge entry by domain experts (based on GUI, complex object cloning, and object-oriented template library),
- Knowledge repository parallel development by distributed teams (based on automatic change propagation and conflict negotiation),
- Knowledge sharing by knowledge maps (based on reusable ontology theory and distributed constraint satisfaction technology), and
- Distributed uncertain knowledge management (based on object-oriented fuzzy dynamic constraint networks as a shared ontology paradigm).

The multi-agent approach is much more suited for scalability than the conventional approach due to its, (a) orientation towards object-oriented modelling for encapsulation, and (b) suitability for unstructured knowledge problem domains. Therefore, a multi-agent approach based on a contract net protocol model has been adopted for our approach. The contract net protocol is a classic co-ordination technique that is used for task and resource allocation between agents. In this protocol agents can play two different roles: a manager and a contractor.

According to FIPA it is necessary to develop the following technological agents: *Facilitator, Mediator, Wrapper* [4]. Furthermore, application-oriented agents have to be implemented. In our approach we see four types of agents:

- *Translation agent* and *Knowledge Fusion (KF) Agent* provide operation performance for KF. *KF* is defined as integration of knowledge from different sources (probably heterogeneous) into a combined resource in order to complement insufficient knowledge and obtain new knowledge.
- *Configuration Agent* supports effective use of knowledge source network, e.g. for integration of knowledge source repositories.
- *Semantic Net Management Agent* provides semantic net operation performance.
- *Monitoring Agent.* Life cycle of rapid knowledge fusion systems consists of forming problem domain (preparation phase) and utilizing it with possible modification (operation phase). During the operation stage rapid knowledge fusion systems work in real-time mode. Accessibility and consistency of knowledge sources are the critical factors for them. Monitoring Agent reduces system failure probability by knowledge source verifications.

Figure 2-3 summarizes the conceptual approach for KSNets in agile SME-networks based on multi-agent technology.

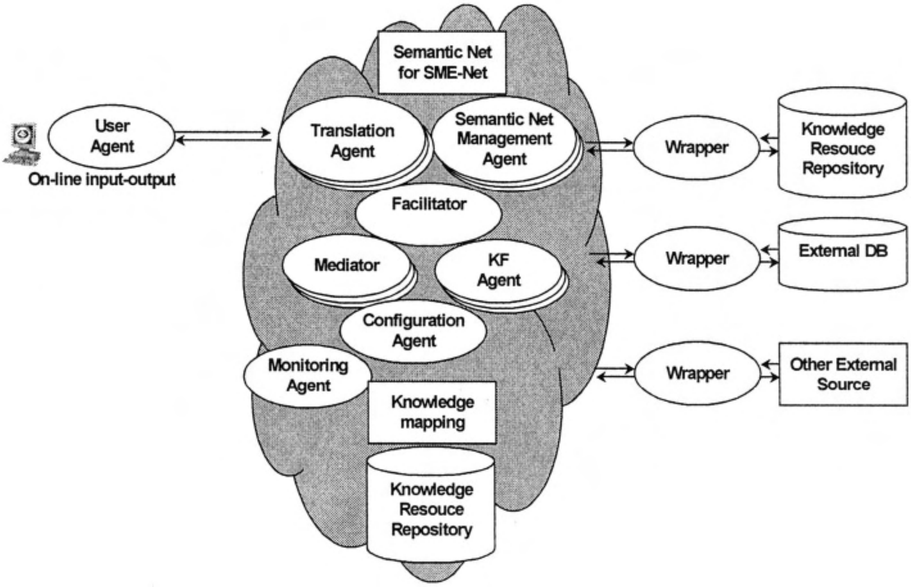


Figure 2-3. General multi-agent framework for KSNet in agile SME-Networks

4. CONCLUSIONS

In our approach, we consider competence models as a knowledge source in SME networks and we use knowledge supply networks as an infrastructure for knowledge logistics. We introduce organizational and technological aspects of our approach including university – SME cooperation, semantic nets and multi-agent framework. Although implementation of the complete approach is still work in progress, first experiences with some parts of the approach have been made.

With respect to the organizational support of SMEs for competence modeling, some early experience was gained in 1991-94 when the ESTPID program [5] was active based on existing and extended cooperation between the electronics departments of the Royal Institute of Technology Stockholm and Tallinn Technical University as a catalyst for developing Estonian Electronic Industry. The results of this project are approx. 30 new jobs in Estonian Electronics Industry, establishing of one new company and creation of different industrial joints.

Furthermore, the approach to competence modeling presented is closely related to the R&D project “SME-Chains” funded by the European Union within the ALFA program. In SME-Chains, four universities from Europe

(Sweden, Norway, Germany, Spain) and five partners from South America (Chile, Argentina, Uruguay, Columbia) establish regional networks of SMEs in their countries in the field of software production. Formation of SME-networks in SME-Chains is based on competence modeling [6]. As a cooperation basis, an Internet-based Web-Portal [2] including competence base has been set up which can serve as a basis for knowledge source repositories.

Experience in multi-agent environments and KSNets has been gathered in several areas, e.g. profile-based configuration of KSNets [15]. Implementation of KSNets is planned for automotive supplier and wood-related industries.

REFERENCES

1. Aguirre J.L., Brena R., Cantu F.J. 2001. Multiagent-based Knowledge Networks. *Expert Systems with applications,* 20, 65—75.
2. Billig A., Sandkuhl K., Wendt A. 2000. Basic Support for Evolutionary Web-Portals: XML-based BaSeWeP Approach. *IASTED 2000 conference proceedings,* Las Vegas.
3. Billig A. *et al.* 2003. Mecomp.net – Organizational, Sociological and Technological Aspects of a Community Network in the Field of Education and Employment. *Euromicro conference PDF 2003,* Genova, Italy, January 2003, IEEE Computer Society Press.
4. Foundation for Intelligent Physical Agents. FIPA 98 Specification. Part 12 – Ontology Service, Oct 23, 1998. *http://www.fipa.org.*
5. Henoch, B. 1992. Universities role in development of electronic industry in Estonia after the liberation. *ESTPID symposium,* TTU, Tallinn, May 1992.
6. Henoch B., Sandkuhl K. 2002. Competence Modelling as a Basis for Formation of SME-Networks – The SME-Chains Approach. *Proc. WWDU 2002 conference.*
7. Hieber R., Alard R. 2000. Supply Chain Management - A Survey of Practices and Trends in Swiss Industry. *Proc. IFIP WG 5.7 Working Conference.* Tromsö, Norway.
8. Holsapple C.W., Singh M. 2001. The Knowledge Chain Model: Activities for Competitiveness. *Expert Systems with Applications,* 20, 77-98.
9. Jacobs N., Shea R. 1996. The Role of Java in InfoSleuth: Agent-Based Exploitation of Heterogeneous Information Resources. *Technical report, Microelectronics and Computer Technology Corporation.*
10. Land A., Pigneur Y. 1999. Digital Trade of Human Competencies. *Proceedings 32nd IEEE Annual Hawaii International Conference on System Sciences,* Vol. 5.
11. Lillehagen F., Karlsen D. 1999. Visual Extended Enterprise Engineering embedding Knowledge Management, Systems Engineering and Work Execution. *IEMC'99 - IFIP International Enterprise Modeling Conference,* Verdal, Norway.
12. Preece A. *et al.* 2000. The KRAFT architecture for knowledge fusion and transformation. *Knowledge-Based Systems,* 13, 113-120.
13. Smirnov A. 1999. Virtual Enterprise Configuration Management. *Proc. of the 14th IFAC World Congress (IFAC'99).* Pergamon Press Beijing, China, vol. A.
14. Smirnov, A. 2001. Rapid Knowledge Fusion into the Scalable Infosphere: A Concept and Possible Manufacturing Applications. *Proceedings of the International NAISO Congress on Information Science Innovations.* U.A.E., Dubai, March 17-20.

15. Smirnov A. 2001. Profile-based configuring of knowledge supply networks in the global business information environment. *Proc. 2001 IEEE Systems, Man and Cybernetics Conference.*
16. Technologies for Enterprise Integration, Integrated Manufacturing Technology Rbadmapping Project, *Oak Ridge Centers for Manufacturing Technology,* Oak Ridge, Tennessee, Rev 3.1, Oct., 1999. (URL: HTTP://IMTI21.ORG).

Chapter 3

A MODELLING FRAMEWORK FOR HUMAN RESOURCE-BASED BUSINESS PROCESSES

Jorge Hermosillo Worley, Bernard Grabot, Laurent Geneste, Omar Aguirre

Abstract: The implementation of an integrated information system, like an ERP (Enterprise Resource Planning) or an APS (Advanced Planning Systems) is nowadays a key issue for companies. The problems that can appear during their integration, which is always a difficult task, are nowadays better identified even if their origins are often multiples and complexes. Defining new business processes and coupling systems with users are thus, sources of many problems. The integration of such systems requires a mapping between the processes used in the company and the standard processes supported by the information systems. In that purpose, a BPR (Business Process Reengineering) project always precedes the implementation of the information system. We suggest to better adapt the business processes to the human actors by explicitly taking into account concepts like the role, competence and knowledge of the human resource. We show how these concepts may complement techniques like BPR in order to better identify the needs and possibilities of the workforce, with the final goal of increasing the efficiency and acceptability of the system to be implemented.

Key words: BPR, human resources, competences, roles, knowledge, ERP.

1. INTRODUCTION

The implementation of an Integrated Information System, like an ERP (Enterprise Resource Planning) or an APS (Advanced Planning and Scheduling) is of vital importance for the companies, but is still a hazardous project: some authors consider that between 25 and 50% of the implementations of ERP systems fail [19]. In all the cases, the implementation of such huge information systems is a complex and risky exercise (see e.g. in [5]).

A key point of the implementation is the mutual compliance of the company and software. Even if it was considered some years ago that software had to be adapted to the company in which it was installed, it is now recognised that evolving in order to be compliant with consistent business processes is a good way to improve the performance of a company. In that purpose, a process modelling project is usually launched at the beginning of the implementation phase, aiming at optimising the business processes of the company first, then at making these optimised processes and the standard software processes consistent. Methods like the BPR method (Business Process Reengineering, [10]) can be used in that purpose.

Some problems which are often occurring during the process modelling phase are shortly described in Section 2. In our opinion, the adaptation of the decisional processes to their human actors can be improved by better taking into account important characteristics of the decision, like the role, competences and knowledge of human resources. A data model which links these characteristics to the BPR concepts is suggested in Section 3. The classification of each concept is proposed in Section 4, and its implementation in a database application is described. The possible uses of this application are suggested in Section 5, and the first industrial applications which are now being performed are shortly described.

2. GENERAL GUIDELINES

Hammer and Champy [10] define BPR as *"the fundamental rethinking and radical redesign of business processes to achieve dramatic improvements in critical, contemporary measures of performance, such as cost, quality, service, and speed"*.

Many very different reasons can be listed in order to understand the difficulty in implementing an ERP [9]. The business process modelling phase clearly participates to these difficulties. According to Yogesh [22], 70% of the BPR projects could be considered as "failures". Some of the causes identified in the literature are:

- For Davenport [4], a design of the processes according to a pure "top down" approach, which is not realistic. The identification and participation of the people to be involved in the BPR project is an essential condition of acceptance and success. This leads to the importance of the correct choice of project teams or groups.
- Another cause of failure is the lack of realism of the objectives and the schedule fixed before the setting of the process reengineering. Indeed, it seems that some managers consider the application of the BPR method as

a tactic, and do not set a real emphasis on its strategic consequences [22]. The BPR is so poorly prepared and the performances to be reached are overestimated.

- Finally, BPR is sometimes applied to cases or processes that do not need changes.

In all these points, it can be noticed that the causes of failures are primarily "human" errors of appreciation and evaluation, and are not linked to the BPR itself. Indeed, these failures are symptoms of an incorrect analyse of the current state of the company. This analyse is one of the main phases of BPR process as stated by Davenport and Short, who suggest a five-step approach to BPR [6]:

a)Develop the Business Vision and Process Objectives: BPR is driven by a business vision which implies specific business objectives such as Cost Reduction, Time Reduction, Output Quality improvement, QWL/Learning/Empowerment [16, 17].

b)Identify the Processes to be Redesigned: Most firms use the *High-Impact* approach which focuses on the most important processes or those that conflict most with the business vision. Less firms use the *Exhaustive* approach that attempts to identify all the processes within an organisation and then prioritise them in order of redesign urgency,

c) Understand and Measure the Existing Processes: For avoiding the repetition of old mistakes and for providing a baseline for future improvements.

d)Identify IT Levers: Awareness of IT (Information Technology) capabilities can and should influence process design.

e)Design and Build a Prototype of the New Process: The actual design should not be viewed as the end of the BPR process, but as a prototype, with successive iterations.

While Hammer and Champy [10] specifically warn against spending too much time studying the current process (steps b and c), European companies usually set the emphasis on the analyse of their present business processes before optimisation. In order to address the problem of the adaptation of the identified process to their actors, we suggest a modelling framework which is described in next section.

3. MODELLINGF RAMEWORK

The interest of modelling techniques for analysing the organisation of information/decision systems in companies is known for a long time. Many methods have been developed in the 80's (and even before) in various areas like production management (GRAI), data structure (MERISE, Entity-Relationship, IDEF3...) or software development (OMT, UML, etc.), see for instance [21] for a tentative review. Efficient process-oriented modelling methods have also been suggested more recently, like ARIS [18]. These methods have often been combined in order to increase their modelling areas, but they have usually been applied in order to assess the consistence of a system or a process, and not (with the notable exception of the GRAI method) in order to better analyse the environment of the decision making activities. In parallel, many works in the area of Human Resource Management have suggested modelling frameworks for the workforce competence [8, 11]. Nevertheless, these works are seldom related to a process view, and mainly focus on technical competences since they are used in order to describe the role of operators, and not of decision makers.

The modelling framework that we suggest is centred on the following concepts:

- *Competences:* In current language, a competence corresponds to a thorough recognised knowledge, which confers the right to judge or decide in certain circumstances. Furthermore, the competence results of the mixed implementation of knowledge, know-how, abilities, attitude and behaviour. More precisely, it has two relevant meanings: the first one addresses the ability of an individual to perform an activity in a job-relevant area. The second one is a definition of what is required from an individual for an effective performance.
- *Role:* The role could be defined as a group of functions to achieve, based on the applications of competences [12]. The role of human beings in the enterprise is fundamental, since people will always make the final decisions. Using the right person at the right moment in the right activity of the process guarantees its efficiency, so that the quality and relevance of the information produced in the process.
- *Knowledge:* Knowledge is a fluid mix of framed experience, values, contextual information, and expert insight that provides a framework for evaluating and incorporating new experiences and information. In organisations, it often becomes embedded not only in documents or repositories but also in organisational routines, processes, practices and norms [7]. In order to optimise the BPR results, we must be able to identify which are the people or resources that consumes and produces the information and knowledge, but also which is the source and type of knowledge.

These concepts are related with the process modelling principles as shown in the general model of Figure 3-1.

On this model, it can be seen that we propose to distinguish between competences *required* by an activity and/or *gained* by the actor. Each of these categories has various types of basic competences which are described in the next section. The actor uses several “informational resources” which allow him to achieve his role. These resources are divided into three categories, namely data, information and knowledge which are also explained in Section 4. Finally, the role definition is based on the application of competences, which are always related to a specific activity including one or more tasks which belong to a given process, with a mission to achieve.

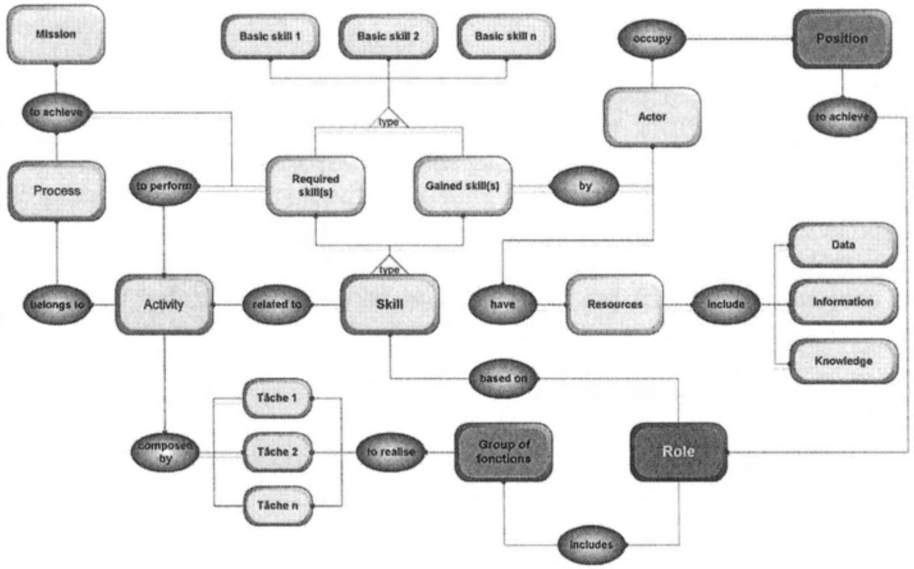


Figure 3-1. General model of the suggested framework

In order to implement this modelling framework in industrial applications, we propose in the next section a general classification of roles, competences and knowledge.

4. CLASSIFICATION AND IMPLEMENTATION OF ROLES, COMPETENCES AND KNOWLEDGE CONCEPTS

4.1 Classification

The following typologies have first been theoretically defined in [12]. They have then been adapted and refined using the experience feedback of the first industrial tests.

4.1.1 Roles

There are various models found in the literature [2, 3, 14, 15] that describe various classes of roles. Each of these models captures important and relevant roles which add further support to the relevance of our classification. Adapted from these works and evolved from the first real applications, we identify four generic classes of roles which could be found in any kind of organisation:

a)The interpersonal roles rise from the hierarchical (and authority) position of the actor in the organisation.

- *Symbol:* Symbolic figure in the organisation, performs duties of social and legal character. Attends to presentations and other activities associated with his/her hierarchical position.
- *Connection:* Establishes contacts with managers and specialists of other divisions and organisations, informing the group of these contacts. Treats business correspondence, and takes part in meetings with representatives of other divisions (organisations).
- *Leader:* In charge of the motivation and the activity of its subordinates. Proactively builds and aligns stakeholders, capabilities, and resources for getting things done quickly and achieving complexes objectives. Ensures that everyone has a useful role and also that the team or group works towards common and agreed goals. Supports its subordinates in their strengths and should use diplomatic competences to overcome conflict.

b)The informational roles are related to information flow, constituting the central node of the organisation. All these roles have an interactive component, but information is not only used to make decisions.

- *Monitor:* The monitor emerges like a central node of internal and external information in the organisation. Searches and integrates information from all sources to develop a well-informed, diverse perspective that can be used to optimise organisational performance.
- *Diffuser:* Transmits information obtained from both external sources and employees to interested people inside the organisation. One part of information relates to facts, the other to interpretation and integration of numerous actions coming from sources of influence for the organisation. It re-directs and verbally transmits information to the others (by means of information letters and digests, etc).
- *Spokesman:* Transmits information of the organisation's plan's, actions, politics, current situation and achievements of the divisions to outsiders and clients (e.g. compiling and disseminating information letters and circulars, participation in meetings with progress reports, etc.).

c)Human resources, by their hierarchical position, have access to certain information which enables them to take part of decision-making processes in the organisation (informational roles).

- *Contractor:* Searches opportunities to develop processes inside the organization and in the interaction with other divisions and structures. Initiates implementation of innovations to improve the organisation's situation and employee well-being. Develops a long-range course of action or set of goals to align with the organisation's vision. Identifies and exploits opportunities for new products, services and markets. Creates an environment that embraces change; makes change happen (even if the change is radical) and helps others to accept new ideas.

- *Regulator:* Integrates corrective actions which should be taken when the organisation faces significant or unexpected disturbances. Assumes responsibility when factors threatening normal work of the organization emerge. Clearly and quickly works through the complexity of key issues, problems and disturbances to affect actions.
- *Resources Distributor:* Decides about organisational allocation of all kinds of resources. Attracts, develops, and retains talent to ensure that people with the right competences and motivations to meet business needs, are in the right place at the right time. Ensures best use is made of each resource's potential. Draws up and approves schedules, plans, estimates and budgets; controls their execution.
- *Negotiator:* Ensures shareholder value through courageous decision-making that supports enterprise or unit-wide interests. Is responsible for ensuring all worthwhile options are considered and act as an arbiter in event of controversy. Represents the organization in important negotiations, establishes official links between the organization and other companies. Is interested in clients and searches to satisfy their needs. Knows the strengths and weaknesses of the company or group.

d)The operational roles are based on a set of knowledge and know-how, implemented by the human resource by using and coordinating specific technical competences.

- *Expert:* Is the specialist who has knowledge in a particular field, acquired by formation, information and experience. Feeds technical information into group and translates from general into technical terms. Contributes with a professional viewpoint on subject under discussion and provides knowledge and techniques in short supply. The experts are classified according to their types of knowledge, are generally the persons in charge for their professional field and are capable of transmit their knowledge to others. They command support as they know more about their subject than anyone else and make decisions and diagnostics based on in-depth experience.
- *Technical Bond:* Turns concepts and ideas into practical working procedures. Has a practical common sense and tackle problem in a systematic way. Responsible for procedures and practical steps to be taken when the group or experts reach significant decisions. Has good organisational competences and competency in tackling necessary tasks by knowing what is feasible and relevant.
- *Operator:* The role or responsibility of operators consists on learning and understanding the operation of the system, machines and tools to be used in

its functional field, applying specific abilities and technical competences, and following the formal procedures established by the organisation.

These roles describe the situations that any actor can assume in a process, and reflects the general functions of people in the organisation. While individuals are not engaged in all these roles, they could be involved in situations related to more than one role at any given time. Furthermore, the actors are often required to adopt several different roles depending on their hierarchical position in the group, but we think emphasis must be placed on identifying which actor is more capable to assume a given role. We believe that a competency approach is necessary to fully develop and define the interrelations between individuals and roles. Indeed, competences can be thought of as the underlying competences or behavioural building blocks inherent in the situational-based roles. Thus, while roles describe the various contexts in which sets or clusters of competences are applied, competencies describe behaviourally specific competences and abilities that impact effectiveness in those contexts [1].

4.1.2 Competences

The control of the interactions in a process under all their forms - negotiation, production, regulation, execution, etc. - and whatever the type of decision to be taken, requires individual and collective competences. We propose five general competence categories:

a)The *technical* competences are relatively well identified by the organisation and the set of involved actors. These competences are always used at the time of the interaction and application of specific technical knowledge, allowing the use of adaptable work tools to the situation or context. They are classified according to the levels of professional areas in three different families:

- *Technical Knowledge:* Theoretical and practical knowledge acquired or necessary to develop a certain technical activity in a specific professional area.
- *Problem-Solving Methods Competences:* Capacities to use the required methodologies to make a diagnosis (implementing the technical knowledge), in order to solve effectively a given problem.
- *Work Tools Competences:* Capacities to use the suitable tools of work.

b)The objective of the *organisational and decisional* competences is of prime interest since it ensures the continuity of the processes. These competences are central since they allow the preparation and accomplishment of the interactions and especially the distribution of tasks and resources. They allow the mobilisation of technical competences in the right level at the right time, the information and material flows, the transmission of the procedures and rules of action to the actors, and in conclusion the coordination and administration of space, time and networks. Their field includes also the transaction, understanding, evaluation and use of results to make decisions. We suggest to classify these competences as follows:

- *Autonomy:* Capacity to make the own tasks independently until the end, when this is required.
- *Use of procedures and rules:* Capacity to apply, evaluate and control the rules and procedures depending on context and situations.
- *Organisation:* Capacity to prepare and follow the actions or activities in an effective way, by taking account of the constraints.
- *Task management:* Capacity to effectively carry out the maximum number of tasks in the minimum amount of time and with the minimum possible effort.
- *Information transmission:* Capacity to transmit relevant information for the team or group in order to ensure the continuity of processes.
- *Decision making:* Capacity to evaluate the elements of a situation or problem, to anticipate the risks related to the considered solutions, and to determine and engage on the best alternative.
- *Delegation:* Capacity to imply and share the decision-making with others in order to increase the autonomy and the sense of responsibilities.
- *Responsibility:* Capacity to assume all the consequences of its acts and decisions.

c) The *adaptation* competences allow the evolution of the organisation through the reception and adaptation of ideas, the attitude facing unexpected situations and the creativity of actors. We can distinguish between:

- *Broad mind:* Fact of being receptive to new or different ideas and concepts.
- *Adaptability:* Capacity to adjust the working methods and behaviour face to new situations, maintaining the effectiveness.
- *Stress Tolerance:* Capacity to resist unfavourable or difficult situations maintaining the usual efficiency and effectiveness.
- *Creativity:* Capacity to imagine and promote innovations.

d)The *interpretation and formalisation* competences are of great importance since they allow the information capitalisation obtained from external and internal sources. The information interpretation and modelling help to understand and formalise the knowledge and experience of the actors in the organisation, with the purpose of capitalise, share and improve the explicit and tacit knowledge of the company. We suggest to classify these competences as follows:

- *Structured Reasoning:* Capacity to structure and correctly formalise documents and to clearly transcribe the ideas and knowledge.
- *Abstraction and simplification:* Capacity to understand, obtain the significant information and simplify the complexity of problems in order to find adapted solutions.

e) *Human* and *motivational* competences give the direction to improve the capabilities of human resources. They lead to explain and present the individual and collective goals, and to assure the correct understanding of roles and internal relations in the company. These competences allow to motivate and animate people, to reduce the tension in difficult situations, to facilitate the complete participation of all the actors and to guide the collective work. They are the key element for the group dynamics and serve to make contact with customers and clients, to understand them and translate their necessities with the purpose of giving and offering adapted solutions. We can list in this category:

- *Collaboration and cooperation:* Spirit of contributing to the collective performance improvement and capacity to mobilise all the contributions to achieve the group goals.
- *Team conduction:* Capacity to lead, carry out, involve and motivate a group, to develop commitment and engagement.
- *Oral communication:* Capacity to effectively communicate the ideas, concepts, information and knowledge.
- *Customer awareness:* Capacity to identify and satisfy the customer requirements, and to improve the relations with internal and external clients.

In real applications, these general types of competences and competences can be directly related to the role classification described above.

4.1.3 Knowledge

We propose like Davenport and Prusak [7] to draw distinctions among data, information and knowledge. Even if a consensual definition of the three terms can hardly be found in the literature, it seems that a progression between them is universally recognised. We shall consider the following definitions, consistent with [20]:

- *information* is a structured set of *data,* on which has been added a meaning or an interpretation,
- associating *information* to a context in order to define application rules allows to build *knowledge.*

For instance, an inventory level (e.g. *10 parts*) is a *data* which can be turned into *information* as soon as an interpretation linked to a situation is added *(low inventory).* This information may lead to use a knowledge such as *"if the inventory is low, a shortage may happen".*

Inside companies, and more precisely within industrial processes, various types of "informational resources" are requested or produced, either in terms of operational data or information (orders, events, files, documents, databases, etc.), or in terms of actual knowledge (models, algorithms, procedures, methods, decision rules, data related to analysis of synthesis, etc.).

The consistence between these elements, which requires first their clarification, is in our opinion a pre-requisite for modelling realistic processes.

Two major categories of knowledge can be distinguished [14]: explicit knowledge constitutes the "learning" of the company, whereas "tacit knowledge" mainly participates to "know-how". For some authors, tacit knowledge, which cannot be formalised, is the kernel of the cognitive wealth of the company. This knowledge can be spread by collaborative working tools, in which the new information technologies play a major role (intranet, groupware, etc.). This knowledge can be organised through management methods like concurrent engineering, competence management, etc. It can be transmitted through apprenticeship, which is more and more seldom and costly. A part of the tacit knowledge can be explicit but not yet formalised knowledge, like all the experience and know-how of the experts which can be difficult to verbalise and to communicate in a structured way; it is also the knowledge which is buried in documents of all kinds, which cannot be retrieved by a simple consultation and which has to be explicitly "brought to light".

According to this model and classifications, a software application based on the Microsoft Access® database has been developed. It is shortly described in next section.

4.2 Implementation

The application, named COCOROL (COnnaissances, COmpétences, ROLes), is composed of ten tables, which structure is shown in Figure 3-2.

The defined application allows first the user to describe a process as a network of activities (execution or decision activities), these processes being either as-is (description of the existing processes) or to-be (description of the optimised processes). It is then possible to describe in further details the decision activities according to the defined concepts (actors, roles, competences and knowledge). The matching of the as-is and to-be processes can then be done, the application allowing for instance the user and the expert to know:

- which are the different human roles in each process,
- which are the required and available human roles,
- what kind of human competences are required for each activity and available in the company,
- when these resources are used in the process,
- which are the data, information and knowledge sources and location, and so on.

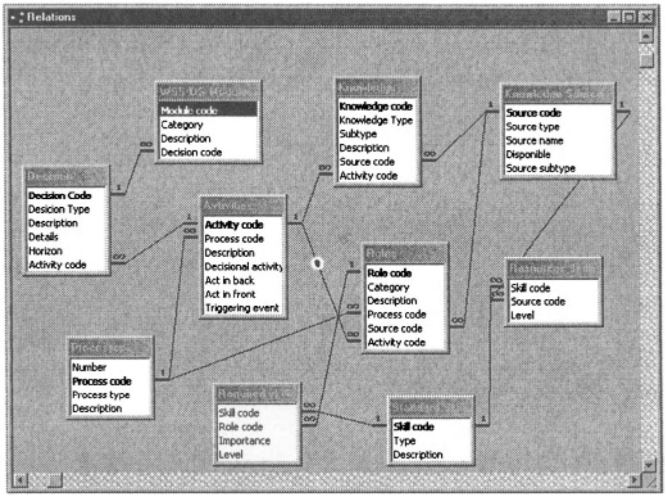


Figure 3-2. Tables and relations of the application

More precisely, the suggested framework and its software implementation may be used in different ways in a process modelling project:

a) Use for the Process Modelling project management: in order to build consistent project teams.

b)Use for process representation: identification of the way the decision makers really work (gained competences, available knowledge).

c)Use for process matching: process from the software, required competences, theoretical knowledge.

The exploitation of the data is achieved by SQL queries using the database environment.

5. REAL LIFE APPLICATIONS

A first comment is that, when we have looked for industrial applications of this work, we have been surprised by the enthusiastic interest of most of the contacted companies. It seems that there is an important industrial expectation on the subject, which is only poorly satisfied by the tools and methods which are available.

Four experiments are at present in progress, showing that different types of applications may be considered for such a framework:

Lesson learnt: A large company of manufacturing railway materials has launched two years ago a project aiming at formalising a lesson learnt process dealing with the expertise on defaults on large components. The objective of the project was primarily to capitalise a knowledge which was essentially ephemeral because of the important turn-over of the experts inside the group. The project has now taken an additional relevance since the version 2000 of the ISO 9001 standard requires to justify the "competence" of the people involved in the processes. Moreover, a major change of version of the ERP used in the company (SAP) should be performed under two years. Within the company, the suggested framework has been used in order to define grids of analysis for supporting interviews of the experts, in order to better identify their level of competence and the knowledge they use during their various activities (analysis of a problem, diagnosis, actions, actualisation of the expertise). This analyse should allow a better "use" of the experts, which constitute a more and more scarce resource in the company (see [13] for more details in this application).

One of the interests of this first experiment is that it fully involves the Human Resource Department, which has provided a substantial help for building up the reference framework for human and motivation competences.

Collection of technical facts: This second experiment has several common points with the first one: its aim is to collect technical facts related to how non-conformance to quality standards is processed, in a company working in the aeronautical area. The company is also engaged in an ISO 9001 version 2000 certification, and intends to model the process of reaction to non-conformity by collecting technical facts using the AVIS tool, available in SAP. The Quality Department considers the project as a first test for validating the interest of a reference framework oriented towards *competence* and *knowledge,* in order to facilitate the introduction of a "continuous improvement" culture in the company. Since problems related to human resources are the main obstacle to such a dynamics, an early identification of the technical competence but also of the personal motivation and brakes of the human actors could be of interest for making this major change easier.

Optimisation ofan ERP system: This project also concerns a company of the aeronautical area located in the south of France, which uses the *BaaN* ERP system. Only a poor analysis of the role and needs of the human resource in the technical processes had been done during the implementation of the system, which has never been accepted by the users. After two years, two major problems can be noticed:

- More and more "parallel" information systems are used in the company. Their users justify their interest by the poor performance of the ERP, but the consequence is that the data in the main information system are not correctly up-dated, which leads to a constantly decreasing performance,
- The number of modules that a user can access in the ERP has become a symbol of influence and power in the company. It has for instance been noticed that each time a decision maker changes of position in the company, he gets access to new modules or screens, but does not accept to

renounce to his rights on the modules he was formerly using. Among the 600 users of the company, more than 250 have now access to all the modules, with a full access to the information contained.

In that difficult context, a major expectation of the ERP manager is to use the modelling framework in order:

- to define the access that a user should have to the information system, on the base of a an objective definition of his role, based on the specification of individual missions, available resources, process objectives and personal competences.
- to better adapt the information system to its users, in order to increase its level of acceptation

Optimisation of service processes: The last project concerns the Panamerican University of Mexico which has just finished the implementation of the *PeopleSoft* ERP. After a phase during which the processes have been successfully modelled, the ERP has been implemented in nominal conditions. Nevertheless, after some months, it seems that the users are not fully satisfied with the adaptation of their workstation to their individual needs. A study has so been launched aiming at complementing the modelled processes with additional information related on one hand to the role and competences required by the activities, and on the other hand to the competences gained by the users. The adaptation of each workstation will as a second step be performed according to the comparison between these two categories of competences.

6. CONCLUSION

The continuous improvement of the organisations is a key challenge for competitiveness, and a specific emphasis has be given during the last ten years on the modelling of efficient business or operational processes, considered as a key factor of success. In that highly competitive context, the workforce of the companies has to constantly adapt its knowledge and behaviour to an ever-changing environment, which is a major cause of stress, human conflicts and resistance to change.

In order to address this kind of problem, we suggest here a general framework allowing a better description of the role of the human resource in the industrial processes by using important concepts like “competence” or “knowledge”. The proposed framework can be used for different applications: quality certification, implementation or optimisation of an ERP system, management of continuous change, optimisation of human resource allocation, optimal forming of project teams, and last but not least, operational management of competences.

In all these cases, there are evidences that the analysis of the role of human resources in correlation to the technical or information processing processes is a key factor for improving the organisation. The results of the four ongoing experiments should allow us to assess the real applicability and interest of the suggested framework in both an industrial and service contexts.

REFERENCES

1. Appelbaum L., Paese M. and Leader P. 2002. *What Senior Leaders Do: The Nine Roles of Strategic Leadership.* White Paper. Development Dimensions International.
2. Belbin R. M. 1981. *Management teams: Why they succeed or fail.* Wiley, New York.
3. Buckinham M. and Coffman C. 1999. *First break all the rules: What the world's greatest managers do differently.* Simon & Schuster, New York.
4. Davenport T.H. 1994. *Reengineering: Business Change of Mythic Proportions?* MIS Quaterly, 121-127.
5. Davenport T.H. 2000. *Mission critical - Realizing the Promise of Enterprise Systems.* Boston, Harvard Business School Press.
6. Davenport T.H. and Short, J 1990. The New Industrial Engineering: Information Technology and Business Process Redesign. *Sloan Management Review,* 31 (4), 11-27.
7. Davenport T.H. and Prusak L. 1998. The Knowledge Creating Company. Working Knowledge, *Harvard Business Review,* November-December, USA.
8. Franchini L., Caillaud, E., Nguyen Ph. and Lacoste, G., 1999. Planning and scheduling competences: towards a human resource management in manufacturing systems, *International Journal of Agile Manufacturing,* 2 (2), 247-260.
9. Grabot B. 2002. The Dark Side of the Moon: some lessons from difficult implementations of ERP systems, *IFAC Ba'02,* Barcelone, July 21-26.
10. Hammer M. and Champy, J. 2001. *Reengineering the corporation: a manifesto for business revolution.* Harper Business.
11. Harzallah M. and Vernadat F., 1999. Human resource competency management in enterprise engineering, *14th IFAC world congress of Information Control in Manufacturing,* Beijing, China, July 5-9.
12. Hermosillo Worley J., Grabot B., Geneste L. and Aguirre O. 2002. Roles, *Competence and Knowledge: Introducing Human Resources in BPR. Proceedings of the ACS – SCM Conference,* Szczecin, Poland. 245-252.
13. Hermosillo Worley J., Rakoto H., Grabot B. and Geneste L. 2003 A Competence Approach in the Experience Feedback Process. *IFIP 2003, International Working Conference of the IFIP WG 5.7: Human Aspects in Production Management,* Karlsruhe, Germany, October 5-9.
14. Hesselbein F., Goldsmith M. and Beckhard R. 1996. *The leader of the future: New visions, strategies, and practices for the next era.* Jossey-Bass, San Francisco.
15. Mintzberg. H. 1979. *The structuring oforganisations,* Prentice Hall.
16. Nonaka I. and Takeuchi H. 1995. *The knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation.* Oxford University Press.
17. Senge P. 1990. *The Fifth Discipline: The Art and Practice of The Learning Organization.* Doubleday Currency, New York.
18. Sheer 1995. *ARIS Toolset documentation.* Saabrücken, Germany.

19. Stewart G. 2001. Stewart G., Factors constraining the exploitation of Enterprise Systems: a research Program, *ACIS 2001, 12th Australasian Conference on Information Systems,* Coles harbour, Australia, December 5-7.
20. Tsuchiya S., 1993. Improving knowledge creation ability through organizational learning, *International Symposium on the Management of Industrial and Corporate Knowledge,* UTC Compiègne, France, October 27-28.
21. Vernadat F. B. 1997. Enterprise Modelling Languages, *International Conference on Enterprise Integration Modelling Technology.* EI-IC ESPRIT Project 21.859, Torino, Italy.
22. Yogesh M. 1998. Business Process Redesign: An Overview. *IEEE Engineering Management Review,* 26 (3), 27-31.

Chapter 4

MANAGING SERVICE-SENSITIVE DEMAND THROUGH SIMULATION

Yuri Merkuryev, Julija Petuhova, Janis Grabis

Abstract: A simulation-based approach for managing supply chains under service-sensitive demand is elaborated. This approach integrates simulation and analytical models. Demand parameters change in response to the short-term service level provided by the supply chain. The simulation model is used for evaluation of the current service level. The analytical models are used to update the parameters of the demand process, which depend upon the current service level, and inventory control parameters. Simulation modelling allows for setting the safety factor at the level ensuring the required long-term service level. Combination of the simulation and analytical models in the runtime regime is vital for modelling the service-sensitive demand.

Key words: supply chain management, runtime simulation, end customer demand, forecasting, uncertain environment.

1. INTRODUCTION

The comparative advantages and disadvantages of analytic versus simulation models are well known (for instance, see Nolan and Sovereign [14]). Hybrid simulation/analytic models are used to attain some of the advantages of both types of models, while avoiding the disadvantages (Shanthikumar and Sargent [18]). Analytic models often have been incorporated into simulation models to solve specific decision-making problems in a cost efficient manner. For instance, analytical models are used to generate demand forecasts (Bhaskaran [6]) and for inventory management (Ganeshan et al. [11]). Shanthikumar and Sargent [18] suggest that it is preferable to obtain values or results required from analytic models prior to simulation instead of continuously using the solution procedure of the analytical models as simulation progresses. However, that limits the variety of interactions the hybrid model is able to represent. The evolution of simulation software and the increase of processing power have led to the opportunity to incorporate analytic models into simulation models in the runtime regime more efficiently. Such an approach is important if outcomes of analytical models depend upon a current state of the simulation execution process.

Alvarez and Centeno [1] describe a simulation model of emergency rooms that had been enhanced with VBA routines, so that it can use real world data. The developed simulation model supports retrieving information at the beginning of a simulation run, while at the same time it allows to change the parameters of the simulation modules at runtime.

Price and Harrell [15] define a runtime simulation regime as a convenient and controlled environment for modifying selected model parameters (capacities, operation times, etc.) without having to change the model data directly. It also provides an experimental environment, which permits multiple scenarios to be defined and simulated.

Clay and Grange [8] simulate the supply chain of automotive service parts in order to evaluate alternative forecasting methods under realistic assumptions. Forecasting methods are continuously invoked during the simulation process. Similarly, Enns [9] investigates the impact of forecasting accuracy on efficiency of materials requirements planning. The author develops a shop floor simulation model, which calls a production scheduling routine in the runtime regime by means of automation.

The runtime regime capabilities are oriented toward helping analysts to change model parameters, incorporate custom procedures (including analytical models), perform simulation runs, and analyse the results of these runs [2]. When a model enters the runtime mode, an analyst can modify characteristics of any objects in the model, including module data, object positions, etc. Generally speaking, it is possible to automate the same actions through external subroutines (using a programming language or some other method) that can be dynamically linked to the simulation model and called from anywhere inside the model at runtime.

We utilize the power of hybrid modelling and interactions between analytical and simulation models in the runtime regime to model inventory management under service-sensitive customer demand. The service-sensitive demand creates dynamic dependencies between itself and supply chain inventory management decisions. These dependencies previously have been investigated in a rather restrictive framework described in the following section. The main modelling objectives are to investigate supply chain behaviour under the service-sensitive demand to propose a mechanism for maintaining the required long-term service level. The simulation model is used to capture supply chain dynamics, while incorporated analytical models are used to model external demand, to forecast external demand and to obtain inventory control decisions. External demand and control decisions are continuously updated according to demand changes in response to the short-term service level provided. The research contributes to the existing body of literature by demonstrating importance of runtime interactions between simulation and analytical models and expanding analysis of inventory management under service-sensitive demand, including development of simulation-based inventory management methods.

The rest of the article is organized as follows. The problem of inventory management under service-sensitive demand is discussed in Section 2. The description of the considered supply chain structure is given in Section 3. Section 4 provides description of the integrated simulation model. Supply chain behaviour is studied in Section 5. Section 6 concludes.

2. PROBLEM BACKGROUND

As markets tend to be more and more customer-oriented, uncertainty connected with the end customer demand and its consequences in the supply chain have become an important subject for research [13]. Inventory control plays an important role in supply chain management. Inventory management involves balancing product availability, or a customer service, on the one hand, with the costs of providing a given level of product availability on the other hand [12]. In fact inventories exist only because supply and demand are not in harmony with each other. Inability to match product availability and customer demand at a given time period leads to either stockouts or extra inventory carrying costs. Stockouts traditionally are accounted for using the liner stockout cost. This stockout penalty, for instance, represents costs associated with extra delivery charges for unavailable products. In other situations, it is used as a proxy variable representing loss of customer goodwill. However, this loss is very hard to measure. Schwartz [17] suggests that the loss of customer goodwill due to stockouts, especially in the retail industry, can be represented more accurately by considering a link between the service level provided and future demand (i.e., customers who face stockouts probably will switch to another retailer, while the high service level is likely to attract additional customers). A traditional formulation of analytic inventory models ignores such dependence. The existing research on inventory management under service sensitive demand has been restricted to either situations with deterministic demand or two-period problems [3], [10]. These limitations can be explained by an explicitly dynamic characteristic of the problem leading to complicated analytic analysis. Simulation modelling allows analyzing a multi-period problem under stochastic demand. The demand parameters change from one period to another in the case of the multi-period problem. The two-period framework defines actions necessary to increase demand by improving the service level. The multi-period framework additionally allows for analyzing consequences of the demand increase (e.g., whether the higher service level can also be maintained). Such behaviour can be observed in highly dynamic and competitive inventory systems (e.g., Silver and Peterson [19]). For instance, wholesalers of computer chips often are not able to meet demand due to insufficient supplies form upstream supply chain levels. In the case of the shortage, customers are likely to seek alternative vendors and may choose to place orders to a newly selected vendor for following periods as long as the service level is maintained. The customers may switch back to the initial vendor, if the newly selected vendor reduces its service level. The research on service-sensitive demand also relates to research on inventory level dependent demand (see Chung [7]) for a recent account) and delivery time sensitive demand (see Ray and Jewkes [16] for instance)

The multiple period inventory management problem under stochastic service-sensitive demand is considered. The service level sensitive demand is modeled similarly to Ernst and Powell [10]. It is updated according to the short-term service level observed during the simulation process. Inventory management parameters are continuously updated according to changes in service-sensitive demand parameters. Forecasting is used to estimate the demand parameters. That allows for controlling supply chain behavior and meeting long-term service level requirements.

3. SUPPLY CHAIN STRUCTURE

The considered supply chain consists of a raw material supplier, a production facility, a finished goods warehouse, and end customers (Figure 4-1).

Here supply and selling points are taken into account accordingly through the external supplier with unlimited capacity and customer demand.

Customers demand a single product from the finished goods warehouse. The finished goods warehouse sees a real customer demand and fulfils customer's orders from the inventory on hand. Inventory control is based on reorder point policy (min-max algorithm) [5] in the finished goods warehouse. If it is necessary to replenish existing inventories, orders are placed to the production facility. Unmet customer demand is lost and hence the customer service level is decreased. Based on the achieved current service level, a future demand forecast is calculated. In accordance with the centralized information strategy concept, a demand forecast is available for the production facility as well.

The production facility produces the quantity of the end product required by the finished goods warehouse.

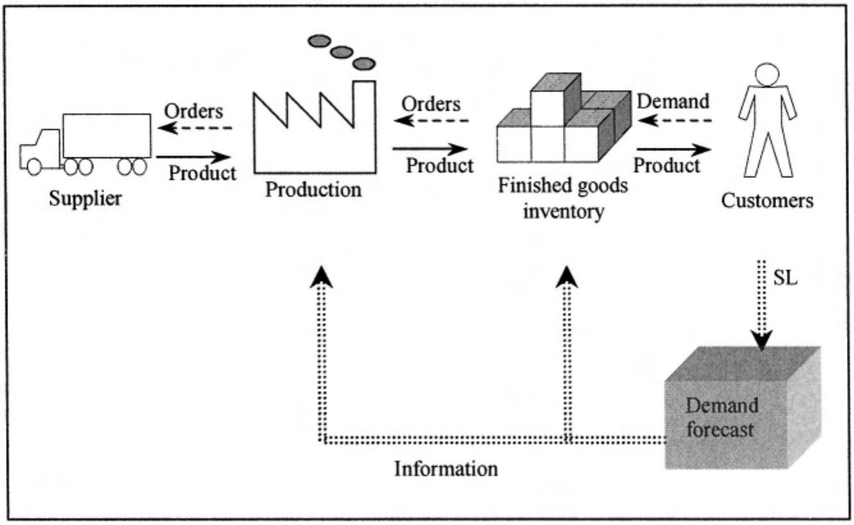


Figure 4-1. The conceptual model of the supply chain

The raw material warehouse is assumed as a part of the production facility or in other words – it is “under the roof of the production. Raw material inventory control is also based on the reorder point policy. The raw material inventory is procured from the external supplier. The external supplier has unlimited capacity (i.e., there are no stockouts). Replenishment quantities for each raw material type are received with a given, planned lead time.

Inventory replenishment and production lead times are planned in this supply chain as well. The inventory replenishment lead time includes the time necessary for order processing and transportation time. The production lead time includes the time necessary to produce certain quantity of finished goods.

The main objective of the entire supply chain is to provide a required long-term service level in the case of service-sensitive demand.

4. INTEGRATED MODEL

Simulation is a powerful tool for analysis of supply chains and inventory systems, because it is capable of capturing the uncertainty and complexity inherent to supply chains and inventory systems. The ability to handle demand and lead time uncertainty is one of the main reasons why simulation is widely used for inventory systems [5].

A structure of the integrated system for supply chain modelling in the case of service-sensitive demand is shown in Figure 4-2. It consists of two parts – a supply chain simulation model and a demand-forecasting model. These two parts interact in the runtime regime.

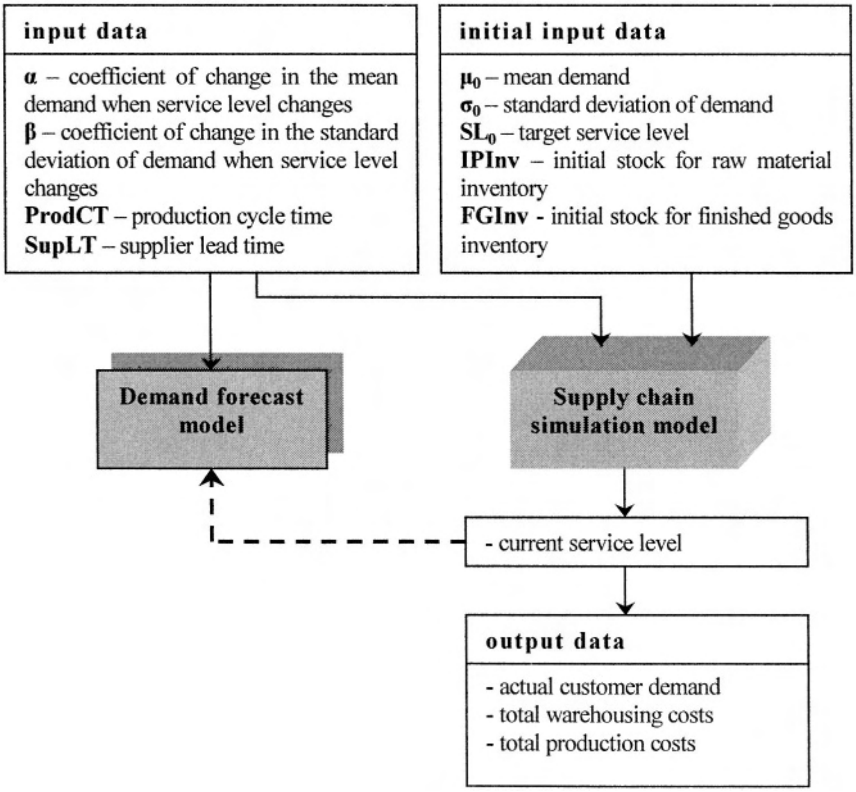

input data
α – coefficient of change in the mean demand when service level changes
β – coefficient of change in the standard deviation of demand when service level changes
ProdCT – production cycle time
SupLT – supplier lead time
initial input data
μ0 – mean demand
σ0 – standard deviation of demand
SL0 – target service level
IPInv – initial stock for raw material inventory
FGInv - initial stock for finished goods inventory
Demand forecast model
Supply chain simulation model
- current service level
output data
- actual customer demand
- total warehousing costs
- total production costs

Figure 4-2. Simulation model framework

The demand forecast model uses the output information of the simulation model (the achieved customer short-term service level) to predict a new end customer demand for the next period. Inventory management decisions made within the simulation model are based on these demand forecasts. Information renovation takes place in a cyclical way within each simulation period. The initial input data block is used to define the starting position for the simulation model. It contains initial statistical information about the demand, received through standard forecasting methods, as well as information about the initial stock level. After the first period these data will be changed in accordance with a new simulation model state. The input data block contains user-defined data. These data are specific for each concrete supply chain and are used during the whole simulation time.

Updating of the demand parameters according to the current state of the system and demand forecasting are key functions in modelling service-sensitive demand. These issues are discussed below.

4.1 Model for Service-Sensitive Demand

Updating and forecasting of customer demand according to the service level provided, implemented as proposed by Ernst and Powell [10], is explored. The main idea of this approach is to develop a model describing dependence of the demand parameters, namely mean and standard deviation, upon changes in the service level.

Universally, it is assumed that the mean demand increases with the service level. The mean demand has a linear relationship with the service level based on the coefficient α that represents change in the mean demand associated with increasing/decreasing the service level. This coefficient is specified based on experience and managerial subjective judgment [10].

Expression (1) describes the relationship between the mean demand and the service level. It implies that the mean demand increases/decreases by α percent for every percentage point increase/decrease in the service level:

$$\mu_{t+1} = \left(1 + \alpha^*\left(SL_t + SL_{t-1}\right)\right)^* \mu_t, \tag{1}$$

where

t – current time period;

μ_{t+1} - mean demand for the next time period;

μ_t - mean demand for the current time period;

SL_t - short-term service level in the current time period;

SL_{t-1} - short-term service level in the previous time period;

α - coefficient of change in the mean demand with increasing/decreasing service level.

The short-term service level is calculated each time period as

$$SL_t = \frac{SO_t}{D_t}, \tag{2}$$

where

SO_t - unsatisfied demand in period *t;*

D_t - observed actual demand in period *t.*

The standard deviation also changes depending upon the service level [10]. If the service level decreases most loyal customers are expected to stay and these customers are likely to exhibit lower demand variability. The service level increase attracts new customers with less established buying behaviour, causing higher variability. The standard deviation of the demand for the new demand level is expressed as a function of the parameters α, β and the current standard deviation σ_t:

$$\sigma_{t+1} = \left[1 + \beta^2 \alpha (SL_t - SL_{t-1})\right]^{\frac{1}{2}} \sigma_t , \tag{3}$$

where
β - coefficient of the change in standard deviation of demand with changed service level.

The coefficient β reflects the difference in variability, if any, between old and new populations. This coefficient is specified in [10] as follows:

- if β is 1, both populations are identical in variability;
- if β is 0, the new demand has no variability;
- if β is 2, the new population has twice the standard deviation of demand of the current population.

The amplitude of changes of the demand parameters are restricted because the market is competitive and other players (not modelled here) will attempt to prevent any other players from expanding their market share substantially.

Here the linear relationship between the service level and the mean demand is assumed. This dependence is evaluated based on parameters estimated by experts. Non-linear relationships are possible in other situations.

4.2 Supply Chain Control Procedure

The considered supply chain simulation model includes analytical inventory management models. These models ought to satisfy the end-customer demand, determining the optimal order-up-to level of each inventory in the supply chain, taking into consideration the predicted service-sensitive customer demand.

It is assumed that the finished goods warehouse and the raw material warehouse (that is a part of the production facility) use the reorder point policy implemented as the min-max algorithm. All stockouts in the finished goods warehouse lead to lost sales and service level decrease. The probability of providing ordered goods from the stock, or item fill rate, is referred to as the long-term service level, and is calculated as

$$SL = \frac{\sum_{t=1}^{T} SO_t}{\sum_{t=1}^{T} D_t}, \tag{4}$$

where
T– total simulation time.

The order size is calculated as the difference between the target quantity and the inventory position. The inventory position is the quantity on hand plus the quantity on order. The order is placed when the inventory position drops below the reorder point (7). The order size can differ from the economic order quantity *(EOQ)* (6) because the amount by which the inventory position drops below the reorder point is added to the *EOQ.*

$$Max_t = ROP_t + EOQ_t, \tag{5}$$

where
Max_t – target inventory quantity in the time period *t;*
ROP_t – reorder point in the time period *t;*
EOQ_t – economic order quantity in the time period *t.*

$$EOQ_t = \sqrt{\frac{2 * \hat{\mu}_{t+1} * OrderCosts}{HoldingCosts}}, \tag{6}$$

$$ROP_t = \hat{\mu}_{t+1} * LT + z * \hat{\sigma}_{t+1} * \sqrt{LT}, \tag{7}$$

where
$\hat{\mu}_{t+1}$ – forecasted mean demand in the previous time period;
$\hat{\sigma}_{t+1}$ – forecasted standard deviation of the mean demand in the previous time period;
LT– replenishment lead-time period;
z – safety factor to ensure the desired probability of not being in stock during the lead time *(LT).*

In the case of normally distributed lead-time demand and steady demand parameters, the safety factor can be found using the standard methods. However, these methods are not valid in the case of service-sensitive demand. Simulation is used to determine the safety factor providing the long-term service level required.

The parameters of demand process and inventory control are periodically updated in accordance with the achieved customer service level. As a result, the initial demand probability distribution will be changed in the runtime regime and the inventory control parameters will be updated accordingly.

5. ANALYSIS OF SUPPLY CHAIN BEHAVIOUR

The objective of experimental studies is to determine, through simulation, the safety stock factor z required to maintain the specified long-term service level. The simulation model is developed using the ARENA® simulation modelling environment. Evaluation of the short-term service level and updating of the demand parameters and the inventory control parameters are implemented using Visual Basic.

A set of experiments with the feedback from the simulation model to the forecasting procedure, when the demand parameters are updated taking into consideration the observed short-term service level (service-sensitive demand), is performed. Performance of the supply chain is evaluated under different safety stock factor values that vary in range between –1 and 1. Other factors such as initial end customer mean demand and its standard deviation, and lead times values, are constant for all experiments.

The design consists of 15 experimental cells. The model was run for 10 replications. Each replication length is defined as 2000 time units and warm-up period is 10 time units.

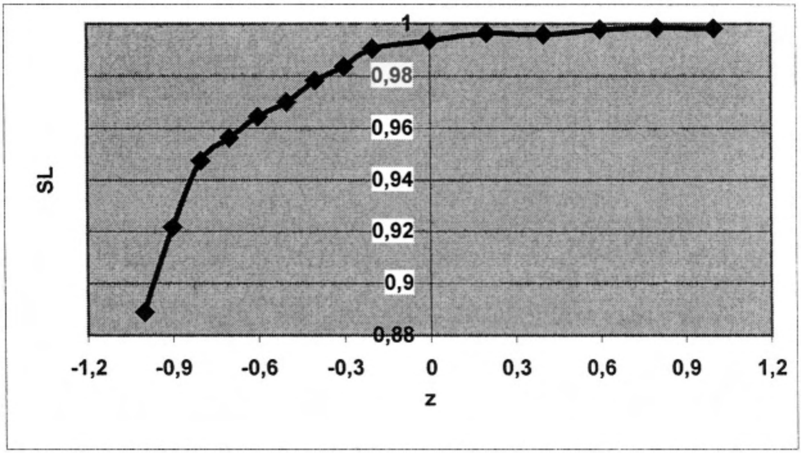


Figure 4-3. Long-term service level dependence from the safety stock factor

The simulation results are averaged over all replications and its graphical representation for one particular set of supply chain parameters is shown in Figure 4-3. Based on the achieved results, it is possible to determine the value of the safety factor needed in order to obtain a specified long-term service level in the supply chain with service-sensitive demand. Safety stock requirements grow relatively quickly if the service level required increases from 0.9 to 0.99. Resolution of the simulation-based procedure is insufficient to accurately set the safety factor if the required service level is above 0.99. The standard method for setting the safety factor in the case of constant demand parameters gives slightly higher safety stock levels. If applied for the service-sensitive demand case, the higher safety stock level leads to unsustainable demand increase. Characteristics of the service level versus safely factor curve depend upon particular values of the supply chain parameters.

A fragment of the dynamic behaviour of the end customer demand and the observed short-term service level during simulation is shown in Figure 4-4. The results are obtained for the safety stock factor equal to –0,8 that should provide a long-term service level of 95% and initial mean demand – 250 units with the standard deviation of 25.

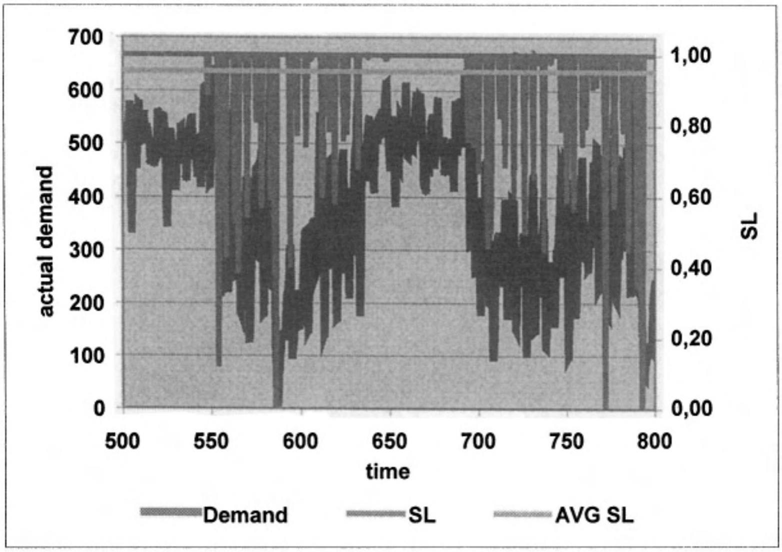


Figure 4-4. End customer demand with target service level 0.95

The results show that the end customer demand increases, if the short-term service level is high, and decreases when the service level becomes lower. This effect is due to differences between the targeted service level and the short-term service level. However, the safety factor level established by means of simulation allows achievement of the long-term service level required.

6. CONCLUSIONS

The simulation-based approach to managing supply chain inventory in the case of service-sensitive demand is developed. It is proposed to integrate the supply chain simulation model and the analytical end customer demand-forecasting model in the runtime regime. As a result, the parameters of the end customer demand are forecasted, taking into consideration the customer service level achieved. The simulation based procedure for setting the safety factor to ensure obtaining the required long-term service level has been explained in detail.

The results obtained demonstrate that integration of simulation and analytical models in the runtime regime is vital for supply chain modelling under the service-sensitive demand. Extension of analysis of service-sensitive demand to multi-period situations reveals complexities of maintaining the improved service level because a larger and more variable population of new customers increases the risk of short-term service level decline.

ACKNOWLEDGMENTS

This research is supported by grant 01.0864 "Development and application of simulation-based tools for analysis and optimisation of logistics systems" from the Latvian Council of Science.

REFERENCES

1. Alvarez A. and Centeno M., 1999. Enhancing simulation models for emergency rooms using VBA. In: *Proceedings of the 1999 Winter Simulation Conference* (WSC'99), Phoenix, Arizona, 5-8 December 1999, 1685-1694.
2. *Arena– Version 7.00,* Rockwell Software Inc., Help Topics: Arena
3. Baker R.C. and Urban T.L., 1988. A deterministic inventory system with an inventory-level-dependent demand rate. *Journal of the Operational Research Society,* 39 (9), 823-831.
4. Ballou R.H., 1999. *Business logistics management.* Prentice-Hall International, Inc., 4th edition, USA, 1999.
5. Banks J. and Malave C.O., 1984. The simulation of inventory systems: An overview. In: *Simulation Councils, Inc.,* June 1984, 283-290.
6. Bhaskaran S. 1998. Simulation analysis of a manufacturing supply chain. *Decision Sciences,* 29 (3), 633-657.
7. Chung K.-J., 2003. An algorithm for an inventory model with inventory-level-dependent demand rate. *Computers & Operations Research,* 30 (9), 1311-1317.
8. Clay G.R. and Grange F., 1997. Evaluating Forecasting Algorithms And Stocking Level Strategies Using Discrete-Event Simulation. In: *Proceedings of the 1997 Winter Simulation Conference* (WSC'97), Atlanta, Georgia, 7-10 December 1997, 817-824.

9. Enns S.T., 2002. MRP performance effects due to forecasting bias and demand uncertainty. *European Journal of Operational Research,* 138, 87-102.
10. Ernst R. and Powell S.G., 1995. Optimal inventory policies under service-sensitive demand. *European Journal of Operational Research,* 87, 316-327.
11. Ganeshan, R., Boone T. and Stenger A. J., 2001. The impact of inventory and flow planning parameters on supply chain performance: An exploratory study. *International Journal of Production Economics,* 71 (1-3), 111-118.
12. Merkuryev Y. and Petuhova J., 2001. Simulation of logistics systems: A survey. In: *Scientific Proceedings of Riga Technical University,* 5 (5), RTU 2001, 125-135.
13. Merkuryev Y., Petuhova J., Van Landeghem R. and Vansteenkiste S., 2002. Simulation-based analysis of the bullwhip effect under different information sharing strategies. In: *Proceedings of the 14th European Simulation Symposium & Exhibition* (ESS’02), Dresden, Germany, 23-26 October 2002, 294-299.
14. Nolan R.L. and Sovereign M.G., 1972. A recursive optimization and simulation approach to analysis with an application to transportation systems. *Management Science,* 18 (12), 676-690.
15. Price R. and Harrell C., 1999. Simulation modelling using Promodel. In: *Proceedings of the 1999 Winter Simulation Conference* (WSC’99), Phoenix, Arizona, 5-8 December 1999, 208-215.
16. Ray S. and Jewkes E.M., 2003. Customer lead time management when both demand and price are lead time sensitive. *European Journal of Operational Research,* in press.
17. Schwartz B.L. 1968. A new approach to stockout penalties. *Management Science,* 12 (12), 538-544.
18. Shanthikumar J.G. and Sargent R.G., 1983. A Unifying View of Hybrid Simulation/Analytic Models and Modeling. *Operations Research,* 31 (6), 1030-1052.
19. Silver E.A. and Peterson R., 1985. *Decision systems for inventory management and production planning.* 2nd ed., New York: John Wiley & Sons, 1985.

This page intentionally left blank

Chapter 5

ROBUST MODELING OF CONSUMER BEHAVIOUR

Maxim Pashkevich, Alexandre Dolgui

Abstract: A new model for describing the probability of customer respond in database marketing is proposed. For this model, new robust estimation and forecasting algorithms are developed. The efficiency of the proposed methods is verified via tests on real-life data from mediaplanning.

Key words: consumer behaviour, beta-binomial model, estimation, forecasting, robustness.

1. INTRODUCTION

Modelling of consumer behaviour is a central problem of database marketing. A key question that has to be answered is which customers in the database to target when selling a particular product. A common approach to this problem is to use the recency, frequency and monetary value (RFM) of past responses to estimate the likelihood of a future response. Having this likelihood and some estimate of the customer profitability, it is possible to segment the database and to choose the optimal group of customers to target [15].

Since the customer response is binary (buy / no by), logistic regression and tree-based regression methods are often used to model the relationship of RFM to response [7]. Another approach uses neural networks to discover relationships between response and behavioural, demographic and other predictors [17]. But although these approaches seem to predict quite well in practice, they suffer from two major drawbacks. First, models that are designed to predict well often provide poor explanation of the nature of the process – an inability to distinguish correlates from drivers [13]. Second, regression models may be thought of as smoothing techniques that attempt to describe the relationship between the predictors and the response but tend to treat heterogeneity as noise. However, taking heterogeneity into account can lead to more accurate prediction [5].

An alternative way to develop a behavioural model that uses observations of past responses to predict future responses is using mixture distributions [6]. Each customer's buying behaviour is characterized with two probability distributions: one for the probability of purchase and one for the monetary amount the customer spends on an individual purchase. Both probabilities have random parameters with known distributions that stand for the heterogeneity in the group of customers. After the parameters of the model are estimated, Bayes forecasting methods are used for prediction [3].

The probability of purchase is usually modelled using beta-binomial distribution [3]. To estimate the parameters of the beta-binomial model, researches usually use classical methods such as method of moments, method of maximum likelihood, χ^2 method and Bayes estimator [16]. However, in practice the theoretical model can be quite different from the real one due to distortions of some kind [8, 9, 11]. As a result, estimation methods that were mentioned above can lead to inconsistent estimates and Bayes predictor can loose its optimality in sense of mean square forecast error.

In this chapter, a new distorted beta-binomial model for describing the probability of customer respond is proposed. For this model, new robust estimation and forecasting algorithms are developed. The efficiency of the proposed methods is verified via tests on real data in mediaplanning.

The chapter is organized as follows. Section 2 contains the mathematical model of consumer behaviour and the forecasting problem statement under distortions. Section 3 is devoted to parameters estimation of the new model of consumer behaviour. In Subsection 3.1, new distorted beta-binomial distribution is introduced. In Subsection 3.2, it is shown that the classical estimation methods for the beta-binomial model can not be applied to estimate the parameters of the new model due to binary data distortions. In Subsections 3.3 and 3.4, new estimators for the cases of both known and unknown distortion levels are proposed. Section 4 is devoted to forecasting of consumer response probabilities. In Subsection 4.1, it is shown that the classical Bayes predictor for the beta-binomial model looses its optimality under distortions. In Section 4.2, new predictor that provides the minimum mean square error in the case of binary data distortions is proposed. Section 5 presents the results of computer simulation that demonstrate the efficiency of the developed robust estimation and forecasting techniques. Section 6 contains the numerical example that illustrates the advantages of the proposed methods via tests on real data from mediaplanning. Section 7 concludes the chapter.

2. PROBLEM STATEMENT

Let us assume that the probability that the customer i responds is p_i, the unit contribution of the customer is μ_i, and the cost of contacting the customer is c_i. Then the expected contribution of the customer is $p_i \cdot \mu_i - c_i$, and the customer should be contacted if $p_i \cdot \mu_i > c_i$. Let the number of potentially interesting customers be K and let us assume that μ_i and c_i are known for each customer. The problem of estimating μ_i and c_i is described in details in [3]. In this work, we concentrate on predicting the response probability p_i.

Suppose that the customer i, with a true unobserved response probability p_i, has been observed to respond r_i times to m_i offers. Assuming that p_i is constant from offer to offer, the distribution of the number of responses to m_i offers by the customer is given by the binomial distribution:

$$P\{r = r_i | m_i, p_i\} = C_{m_i}^{r_i} p_i^{r_i} (1 - p_i)^{m_i - r_i}. \tag{1}$$

A simple estimate of p_i is just r_i/m_i. However, the problem with this estimate is that in practice m_i is usually too small. Aggregation across all customers to estimate p as

$$\hat{p} = \sum_{i=1}^{K} r_i \Big/ \sum_{i=1}^{K} m_i \tag{2}$$

will only be appropriate if all customers have the same response probability, and it is not possible because of heterogeneity among customers. A way out is to specify a probability distribution for the response probability, so let us assume that the p_i's have a beta distribution with true unknown parameters α_0, β_0:

$$f_p(x | \alpha_0, \beta_0) = x^{\alpha_0 - 1}(1 - x)^{\beta_0 - 1} / B(\alpha_0, \beta_0). \tag{3}$$

The beta distribution is very flexible and can take a variety of shapes depending on the values of the parameters α, β [10]. For example, if α and β are both less than 1 the distribution will be U- or J-shaped. This shape represents a polarized distribution where some consumers have small response probabilities and others have large response probabilities, but few customers are in between. On the other hand, if α and β are both large, the distribution will resemble a spike so that all customers have more or less the same response probability. Values of α and β just a little larger than 1 make the beta distribution look like an inverted U or like the central part of the normal curve.

Under the assumption that the p_i's are beta distributed, the distribution of the number of responses to m_i offers by the customer is given by the beta-binomial distribution with parameters m_i, α_0, β_0:

$$P\{r = r_i | m_i, p_i\} = C_{m_i}^{r_i} \, B(\alpha_0 + r_i, \beta_0 + m_i - r_i)/B(\alpha_0, \beta_0). \qquad (4)$$

Beta-binomial distribution (BBD) is widely used in marketing for modelling brand choice of consumer goods [12], in advertising for modelling reach and frequency [14], and in direct marketing for modelling list fall-off [1]. Bayes prediction of the response probabilities based on the beta-binomial model looks like

$$E\{p_i | r_i, m_i, \hat{\alpha}, \hat{\beta}\} = (\hat{\alpha} + r_i)/(\hat{\alpha} + \hat{\beta} + m_i), \qquad (5)$$

where $\hat{\alpha}$, $\hat{\beta}$ are estimates of the corresponding parameters of the model.

Let us now have information about n campaigns (selling offers, advertisements, etc.) in the past and let $B = (b_{ij})$ be a binary $K \times n$ matrix where $b_{ij} = 1$ if the customer i responded in the campaign j after he/she was contacted, and $b_{ij} = 0$ otherwise. Let us note that in this case $m_i = n$, $\forall\ i = 1, 2, \ldots, K$. We observe the distorted binary matrix $\widetilde{B} = (\widetilde{b}_{ij})$:

$$\widetilde{b}_{ij} = b_{ij} \oplus \eta_{ij}, \quad P\{\eta_{ij} = 1\} = \begin{cases} \varepsilon_0, \, b_{ij} = 0; \\ \varepsilon_1, \, b_{ij} = 1; \end{cases} \qquad (6)$$

where $\oplus$ is a binary addition operator, $\{\eta_{ij}\}$ are independent random Bernoulli variables, and ε_0, ε_1 are distortion levels. The case when $\varepsilon_0 = \varepsilon_1 = 0$ corresponds to the situation when no distortions are present. This distortion model was proposed by Copas and is widely used in practical applications [4].

The problem is to find estimates of the parameters α, β having the distorted sample $X = (x_1, x_2, \ldots, x_K)$ of size K:

$$x_i = \sum_{i=1}^{n} \widetilde{b}_{ij}, \quad i = 1, 2, \ldots, K, \qquad (7)$$

and to predict the unknown probabilities p_i, $i = 1, 2, \ldots, K$, that characterize customers respond. Here x_i is the number of times the customer i responded in the past n campaigns.

3. ROBUST ESTIMATION OF THE PARAMETERS OF THE CONSUMER RESPONSE MODEL

In case when there are no distortions present, the distribution of the sample X is the BBD with the parameters n, α_0, β_0, where α_0, β_0 are unknown. However, it can be proved that under distortions the sample X is described by a different distribution that we call distorted beta-binomial distribution (DBBD) with the parameters n, α_0, β_0, ε_0, ε_1. It is shown that using the method of moments (MM) and method of maximum likelihood (MML) for the BBD to estimate the α_0, β_0 parameters of the DBBD leads to inconsistent estimates. For this reason, new methods of estimation of the parameters α_0, β_0 and the distortion levels ε_0, ε_1 are proposed.

3.1 Distorted beta-binomial distribution

It can be shown that the probability distribution of the distorted sample X is given by the following equations (i, r = 0, 1, ..., n)

$$p_r(\alpha,\beta,\varepsilon_0,\varepsilon_1) = P\{x_1 = r\} = \sum_{i=0}^{n} w_{ri}(\varepsilon_0,\varepsilon_1)\cdot C_n^i \frac{B(\alpha_0+i,\beta_0+n-i)}{B(\alpha_0,\beta_0)}, \qquad (8)$$

$$w_{ri}(\varepsilon_0,\varepsilon_1) = \sum_{l=\max(i,r)}^{\min(n,i+r)} C_i^{l-r} C_{n-i}^{l-i} \varepsilon_0^{l-i}(1-\varepsilon_0)^{n-l}\varepsilon_1^{l-r}(1-\varepsilon_1)^{i+r-l}. \qquad (9)$$

Let us call this distribution the DBBD with the parameters n, α_0, β_0, ε_0, ε_1. The first four moments of this distribution are calculated as (m^0, m_3^0, m_4^0 are the 1st, 3rd and 4th moments of the BBD with the parameters n, α_0, β_0)

$$m(\varepsilon_0,\varepsilon_1) = E\{\tilde{\xi}\} = m^0 + (1-m^0)\varepsilon_0 - m^0\varepsilon_1, \qquad (10)$$

$$\begin{aligned} m_2(\varepsilon_0,\varepsilon_1) &= m(\varepsilon_0,\varepsilon_1) + \\ &+ \frac{n^{[2-]}}{(\alpha+\beta)^{[2+]}}\left(\alpha^{[2+]} + \beta^{[2+]}\varepsilon_0^2 + \alpha^{[2+]}\varepsilon_1^2 + 2\alpha\beta\varepsilon_0 - 2\alpha^{[2+]}\varepsilon_1 - 2\alpha\beta\varepsilon_0\varepsilon_1\right), \end{aligned} \qquad (11)$$

$$\begin{aligned} m_3(\varepsilon_0,\varepsilon_1) &= m_3^0 + \left(\frac{n\beta}{\alpha+\beta} + 6n^{[2-]}\frac{\alpha\beta}{(\alpha+\beta)^{[2+]}} + 3n^{[3-]}\frac{\alpha^{[2+]}\beta}{(\alpha+\beta)^{[3+]}}\right)\varepsilon_0 + \\ &+ \left(\frac{n\alpha}{\alpha+\beta} + 6n^{[2-]}\frac{\alpha^{[2+]}}{(\alpha+\beta)^{[2+]}} + 3n^{[3-]}\frac{\alpha^{[3+]}}{(\alpha+\beta)^{[3+]}}\right)\varepsilon_1 + o(\varepsilon_0) + o(\varepsilon_1), \end{aligned} \qquad (12)$$

$$
\begin{aligned}
&m_4(\varepsilon_0,\varepsilon_1)=m_4^0+\\
&+\left(\frac{n\beta}{\alpha+\beta}+14\,n^{[2-]}\frac{\alpha\beta}{(\alpha+\beta)^{[2+]}}+18\,n^{[3-]}\frac{\alpha^{[2+]}\beta}{(\alpha+\beta)^{[3+]}}+4\,n^{[4-]}\frac{\alpha^{[3+]}\beta}{(\alpha+\beta)^{[4+]}}\right)\varepsilon_0+\\
&+\left(\frac{n\alpha}{\alpha+\beta}+14\,n^{[2-]}\frac{\alpha^{[2+]}}{(\alpha+\beta)^{[2+]}}+18\,n^{[3-]}\frac{\alpha^{[3+]}}{(\alpha+\beta)^{[3+]}}+4\,n^{[4-]}\frac{\alpha^{[4+]}}{(\alpha+\beta)^{[4+]}}\right)\varepsilon_1+o(\varepsilon_0)+o(\varepsilon_1),
\end{aligned}
\tag{13}
$$

where $y^{[z-]}=y(y-1)\ldots(y-z+1)$, $y^{[z+]}=y(y+1)\ldots(y+z-1)$, $y\in R, z\in N$.

3.2 Inconsistency of the MM and MML Estimators Under Distortions

The following results that explore the properties of the classical MM-estimators and MML-estimators of the α, β parameters of the beta-binomial distribution were obtained.

Let us have the described above series of experiments under distortions (6) with levels ε_0, ε_1 and let X be the corresponding distorted sample of size K. Then it can be proved that the following asymptotic expansions are true for the MM-estimators of the α, β parameters of the beta-binomial model:

$$\tilde{\alpha}_{MM}(\varepsilon_0,\varepsilon_1)=\alpha_0+(\alpha_0+2\beta_0+1)\,\varepsilon_0+\frac{\alpha_0(\alpha_0+1)}{\beta_0}\varepsilon_1+o(\varepsilon_0)+o(\varepsilon_1), \tag{14}$$

$$\tilde{\beta}_{MM}(\varepsilon_0,\varepsilon_1)=\beta_0+\frac{\beta_0(\beta_0+1)}{\alpha_0}\varepsilon_0+(2\alpha_0+\beta_0+1)\,\varepsilon_1+o(\varepsilon_0)+o(\varepsilon_0). \tag{15}$$

Under the same conditions, the following asymptotic expansions for the MML-estimators of the α, β parameters of the beta-binomial model hold

$$\begin{pmatrix}\Delta\tilde{\alpha}_{MML}(\varepsilon_0,\varepsilon_0)\\ \Delta\tilde{\beta}_{MML}(\varepsilon_0,\varepsilon_0)\end{pmatrix}=\begin{pmatrix}H_{11} & H_{12}\\ H_{21} & H_{22}\end{pmatrix}^{-1}\cdot\begin{pmatrix}G_{11} & G_{12}\\ G_{21} & G_{22}\end{pmatrix}\cdot\begin{pmatrix}\varepsilon_0\\ \varepsilon_1\end{pmatrix}+\begin{pmatrix}o(\varepsilon_0)+o(\varepsilon_1)\\ o(\varepsilon_0)+o(\varepsilon_1)\end{pmatrix}, \tag{16}$$

where

$$\Delta\tilde{\alpha}_{MML}(\varepsilon_0,\varepsilon_1)=\tilde{\alpha}_{MML}(\varepsilon_0,\varepsilon_1)-\alpha_0,\quad \Delta\tilde{\beta}_{MML}(\varepsilon_0,\varepsilon_1)=\tilde{\beta}_{MML}(\varepsilon_0,\varepsilon_1)-\beta_0, \tag{17}$$

$$H_{11}=S_{\alpha\beta}-S_{\alpha},\quad H_{12}=H_{21}=S_{\alpha\beta},\quad H_{22}=S_{\alpha\beta}-S_{\beta}, \tag{18}$$

$$G_{11} = S_{\alpha p}, \quad G_{12} = S^{+}_{\alpha p} = S^{+}_{\beta p} = G_{21}, \quad G_{22} = S_{\beta p}, \tag{19}$$

$$p_j(\varepsilon_0,\varepsilon_1) = P\{x_1 = j\}, \; j = 0,1,\ldots,n, \quad P_i(\varepsilon_0,\varepsilon_1) = \sum_{j=0}^{i} p_j(\varepsilon_0,\varepsilon_1), \tag{20}$$

$$S_{\alpha} = \sum_{i=0}^{n-1} \frac{1 - P_i(0,0)}{(\alpha_0 + i)^2}, \; S_{\beta} = \sum_{i=0}^{n-1} \frac{P_i(0,0)}{(\beta_0 + n - i - 1)^2}, \; S_{\alpha\beta} = \sum_{i=0}^{n-1} \frac{1}{(\alpha_0 + \beta_0 + i)^2}, \tag{21}$$

$$S_{\alpha p} = -\sum_{i=0}^{n-1} \frac{(n-i)p_i(0,0)}{\alpha_0 + i}, \quad S^{+}_{\alpha p} = \sum_{i=0}^{n-1} \frac{(i+1)p_{i+1}(0,0)}{\alpha_0 + i}, \tag{22}$$

$$S_{\beta p} = \sum_{i=0}^{n-1} \frac{(n-i)p_i(0,0)}{\beta_0 + n - i - 1}, \quad S^{+}_{\beta p} = -\sum_{i=0}^{n-1} \frac{(i+1)p_{i+1}(0,0)}{\beta_0 + n - i - 1}. \tag{23}$$

The obtained results allow us to make the conclusion that, in the case of distorted sample, the classical MM-estimators and MML-estimators become inconsistent.

3.3 Robust Estimation in the Case of Known Distortion Levels

In practice, it is a common situation when the estimates of distortion levels are known in advance. The source of prior information can be the past work experience or experts knowledge. In this case, it is possible to develop new methods of the beta-binomial model parameters estimation that lead to consistent estimators of α, β:

$$\hat{\alpha}_{MM}(\varepsilon_0,\varepsilon_1) = \frac{(m^* - \varepsilon_0 n)\left(m^* n - m_2^* + \varepsilon_0(n-1)(m^* - (1-\varepsilon_1)n) - m^*\varepsilon_1(n-1)\right)}{(1 - \varepsilon_0 - \varepsilon_1)\left(m_2^* n - m^* n - m^{*2}(n-1)\right)}, \tag{24}$$

$$\hat{\beta}_{MM}(\varepsilon_0,\varepsilon_1) = \frac{(m^* - (1-\varepsilon_1)n)\left(m_2^* + m^*\varepsilon_1(n-1) - m^* n - \varepsilon_0(n-1)(m^* - n(1-\varepsilon_1))\right)}{(1 - \varepsilon_0 - \varepsilon_1)\left(m_2^* n - m^* n - m^{*2}(n-1)\right)}, \tag{25}$$

where $m^* = K^{-1}\sum_{i=1}^{K} x_i$, $\quad m_2^* = K^{-1}\sum_{i=1}^{K} x_i^2$.

Another way of using the priori information is filtering the given distorted sample X. Let $\tilde{p}(\varepsilon_0,\varepsilon_1)$ be a vector of empirical relative frequencies calculated using the sample X. This vector corresponds to the distorted beta-binomial distribution with parameters n, α_0, β_0, ε_0, ε_1, where α_0, β_0 are unknown. Let $W = (w_{ij})$ be a $(n + 1)$x$(n + 1)$ matrix defined by equation (9) and let the vector $\widetilde{p}_0$ be defined as $\widetilde{p}_0 = W^{-1}(\varepsilon_0, \varepsilon_1) \cdot \widetilde{p}(\varepsilon_0, \varepsilon_1)$. It can be proved that the vector $\widetilde{p}_0$ is a vector of empirical relative frequencies for the beta-binomial distribution with the parameters n, α_0, β_0 where α_0, β_0 are unknown. As a result, all known methods of estimation of the parameters of the beta-binomial distribution [16] can be used now to estimate the true values of α, β.

3.4 Robust Estimation in the General Case

Let us consider a more general case now when the distortion levels ε_0, ε_1 are unknown. Two iterative algorithms of simultaneous estimation of the parameters of the model and of the distortion levels are proposed. Both methods suppose that the sample X is a sample from the distorted beta-binomial distribution.

The first algorithm is based on the method of moments. The idea is to iteratively solve the system of two non-linear equations using the modified method of Newton:

$$m_3^* = m_3(\alpha(\varepsilon_0, \varepsilon_1), \beta(\varepsilon_0, \varepsilon_1), \varepsilon_0, \varepsilon_1), \quad m_4^* = m_4(\alpha(\varepsilon_0, \varepsilon_1), \beta(\varepsilon_0, \varepsilon_1), \varepsilon_0, \varepsilon_1), \tag{26}$$

$$m_3^* = K^{-1} \sum\nolimits_{i=1}^{K} x_i^3, \qquad m_4^* = K^{-1} \sum\nolimits_{i=1}^{K} x_i^4. \tag{27}$$

Here $m_3(.)$, $m_4(.)$ are the 3rd and the 4th theoretical moments of the distorted beta-binomial distribution with the parameters α, β, ε_0, ε_1, and $\alpha(\varepsilon_0,\varepsilon_1)$, $\beta(\varepsilon_0,\varepsilon_1)$ are the estimates of the parameters α, β when the distortions levels ε_0, ε_1 are fixed. These estimates can be calculated using equations (24), (25).

Let J_0^{cond} be a Jacobi matrix of the system (26) under the assumption that the first two moments are fixed: $m = const,\ m_2 = const$. Then using the modified method ofNewton will lead us to the following iterative procedure:

$$\begin{pmatrix} \varepsilon_0^{k+1} \\ \varepsilon_1^{k+1} \end{pmatrix} = \begin{pmatrix} \varepsilon_0^{k} \\ \varepsilon_1^{k} \end{pmatrix} + \lambda \cdot \left(J_0^{cond}\right)^{-1} \cdot \begin{pmatrix} m_3^* - m_3\left(\alpha(\varepsilon_0^k, \varepsilon_1^k), \beta(\varepsilon_0^k, \varepsilon_1^k), \varepsilon_0^k, \varepsilon_1^k\right) \\ m_4^* - m_4\left(\alpha(\varepsilon_0^k, \varepsilon_1^k), \beta(\varepsilon_0^k, \varepsilon_1^k), \varepsilon_0^k, \varepsilon_1^k\right) \end{pmatrix}, \tag{28}$$

where $\lambda \in (0, 1]$ is the algorithm parameter that ensures convergence of the procedure in the case of large distortion levels ε_0, ε_1. Let us introduce the following notation for the partial derivatives under the assumption $m = const,\quad m_2 = const$, $\varepsilon_0 = 0$, $\varepsilon_1 = 0$: $(\partial f/\partial x)_0^c$. Then the elements of the matrix J_0^{cond} are calculated as

$$\left(\frac{\partial m_3}{\partial \varepsilon_0}\right)_0^c = 3n^{[3-]}\frac{\alpha^{[2+]}\beta}{(\alpha+\beta)^{[3+]}},\quad \left(\frac{\partial m_3}{\partial \varepsilon_1}\right)_0^c = 3n^{[3-]}\frac{\alpha^{[3+]}}{(\alpha+\beta)^{[3+]}}, \tag{29}$$

$$\left(\frac{\partial m_4}{\partial \varepsilon_0}\right)_0^c = 14n^{[4-]}\frac{\alpha^{[3+]}\beta}{(\alpha+\beta)^{[4+]}} + 6\left(\frac{\partial m_3}{\partial \varepsilon_0}\right)_0^c,\ \left(\frac{\partial m_4}{\partial \varepsilon_1}\right)_0^c = 4n^{[4-]}\frac{\alpha^{[4+]}}{(\alpha+\beta)^{[4+]}} + 6\left(\frac{\partial m_3}{\partial \varepsilon_1}\right)_0^c, \tag{30}$$

$$\left(\frac{\partial \hat{\alpha}_{MM}}{\partial \varepsilon_0}\right)_0^c = -(\alpha_0 + 2\beta_0 + 1),\quad \left(\frac{\partial \hat{\alpha}_{MM}}{\partial \varepsilon_1}\right)_0^c = -\alpha_0(\alpha_0+1)/\beta_0, \tag{31}$$

$$\left(\frac{\partial \hat{\beta}_{MM}}{\partial \varepsilon_0}\right)_0^c = -\beta_0(\beta_0+1)/\alpha_0,\quad \left(\frac{\partial \hat{\beta}_{MM}}{\partial \varepsilon_1}\right)_0^c = -(2\alpha_0 + \beta_0 + 1), \tag{32}$$

$$\left(\frac{\partial m_3}{\partial \alpha}\right)_0^c = n^{[3-]}\frac{\alpha^{[3+]}}{(\alpha+\beta)^{[3+]}}\sum_{i=0}^{2}\left(\frac{1}{\alpha+i} - \frac{1}{\alpha+\beta+i}\right), \tag{33}$$

$$\left(\frac{\partial m_3}{\partial \alpha}\right)_0^c = n^{[3-]}\frac{\alpha^{[3+]}}{(\alpha+\beta)^{[3+]}}\sum_{i=0}^{2}\left(-\frac{1}{\alpha+\beta+i}\right), \tag{34}$$

$$\left(\frac{\partial m_4}{\partial \alpha}\right)_0^c = n^{[4-]}\frac{\alpha^{[4+]}}{(\alpha+\beta)^{[4+]}}\sum_{i=0}^{3}\left(\frac{1}{\alpha+i} - \frac{1}{\alpha+\beta+i}\right) + 6\left(\frac{\partial m_3}{\partial \alpha}\right)_0^c, \tag{35}$$

$$\left(\frac{\partial m_4}{\partial \alpha}\right)_0^c = n^{[4-]}\frac{\alpha^{[4+]}}{(\alpha+\beta)^{[4+]}}\sum_{i=0}^{3}\left(-\frac{1}{\alpha+\beta+i}\right) + 6\left(\frac{\partial m_3}{\partial \alpha}\right)_0^c. \tag{36}$$

The second algorithm is based on the method of maximum likelihood which leads us to the following problem of conditional maximization:

$$\begin{gathered} l(\alpha,\beta,\varepsilon_0,\varepsilon_1) = \textstyle\sum_{r=0}^{n} f_r \ln(p_r(\alpha,\beta,\varepsilon_0,\varepsilon_1)) \to \max, \\ \alpha > 0,\ \beta > 0,\ 0 \le \varepsilon_0, \varepsilon_1 \le 1, \end{gathered} \tag{37}$$

where $\{f_r\}$ are the frequencies for the distorted sample X. The problem of constrained maximization is solved using the modification of steepest descent method that uses the following expressions for the function $l(.)$ derivatives:

$$\frac{\partial l}{\partial \alpha}(\alpha,\beta,\varepsilon_0,\varepsilon_1)=\sum_{r=0}^{n}\left(f_r\sum_{i=0}^{n}w_{ri}(\varepsilon_0,\varepsilon_1)\,\partial p_i^0(\alpha,\beta)/\partial\alpha\Big/p_r(\alpha,\beta,\varepsilon_0,\varepsilon_1)\right), \quad (38)$$

$$\frac{\partial l}{\partial \beta}(\alpha,\beta,\varepsilon_0,\varepsilon_1)=\sum_{r=0}^{n}\left(f_r\sum_{i=0}^{n}w_{ri}(\varepsilon_0,\varepsilon_1)\,\partial p_i^0(\alpha,\beta)/\partial\beta\Big/p_r(\alpha,\beta,\varepsilon_0,\varepsilon_1)\right), \quad (39)$$

$$\frac{\partial l}{\partial \varepsilon_0}(\alpha,\beta,\varepsilon_0,\varepsilon_1)=\sum_{r=0}^{n}\left(f_r\sum_{i=0}^{n}\partial w_{ri}(\varepsilon_0,\varepsilon_1)/\partial\varepsilon_0\;p_i^0(\alpha,\beta)\Big/p_r(\alpha,\beta,\varepsilon_0,\varepsilon_1)\right), \quad (40)$$

$$\frac{\partial l}{\partial \varepsilon_1}(\alpha,\beta,\varepsilon_0,\varepsilon_1)=\sum_{r=0}^{n}\left(f_r\sum_{i=0}^{n}\partial w_{ri}(\varepsilon_0,\varepsilon_1)/\partial\varepsilon_1\;p_i^0(\alpha,\beta)\Big/p_r(\alpha,\beta,\varepsilon_0,\varepsilon_1)\right), \quad (41)$$

$$\frac{\partial p_i^0(\alpha,\beta)}{\partial\alpha}=p_i^0(\alpha,\beta)\left(\sum_{j=0}^{i-1}\frac{1}{\alpha+j}-\sum_{j=0}^{n-1}\frac{1}{\alpha+\beta+j}\right), \quad (42)$$

$$\frac{\partial p_i^0(\alpha,\beta)}{\partial\beta}=p_i^0(\alpha,\beta)\left(\sum_{j=0}^{n-i-1}\frac{1}{\beta+j}-\sum_{j=0}^{n-1}\frac{1}{\alpha+\beta+j}\right), \quad (43)$$

$$\frac{\partial w_{ri}}{\partial\varepsilon_0}(\varepsilon_0,\varepsilon_1)=\sum_{l=\max(i,r)}^{\min(n,i+r)}C_i^{l-r}C_{n-i}^{l-i}\left((l-i)\varepsilon_0^{l-i-1}(1-\varepsilon_0)^{n-l}-(n-l)\varepsilon_0^{l-i}(1-\varepsilon_0)^{n-l-1}\right)\varepsilon_1^{l-r}(1-\varepsilon_1)^{i+r-l}, \quad (44)$$

$$\frac{\partial w_{ri}}{\partial\varepsilon_1}(\varepsilon_0,\varepsilon_1)=\sum_{l=\max(i,r)}^{\min(n,i+r)}C_i^{l-r}C_{n-i}^{l-i}\varepsilon_0^{l-i}(1-\varepsilon_0)^{n-l}\left((l-r)\varepsilon_1^{l-r-1}(1-\varepsilon_1)^{i+r-l}-(i+r-l)\varepsilon_1^{l-r}(1-\varepsilon_1)^{i+r-l-1}\right). \quad (45)$$

Let us refer to the described above estimation algorithm as maximum likelihood simultaneous estimation (MLSE).

4. ROBUST FORECASTING OF THE CONSUMER RESPONSE PROBABILITIES

In case when no distortions are present, Bayes predictor (5) leads to the minimum mean square forecast error and for this reason is usually used in practical applications. However, it can be proved that in the case of binary distortions (6), the classical Bayes predictor (5) looses its optimality. For this reason, a new Bayes-based predictor that takes into account possible data distortions and provides the minimum mean square error is proposed.

4.1 The Robustness of the Classical Bayes Predictor under Distortions

Let us have the described above series of experiments under stochastic binary distortions with levels ε_0, ε_1 and let α_0, β_0 be the true values of BBD parameters. Then the mean square forecast error of the Bayes predictor (5) is defined as

$$\tilde{r}^2_{bayes} = \frac{\alpha_0\beta_0}{(\alpha_0+\beta_0)^{[2+]}(\alpha_0+\beta_0+\mathrm{n})} + \frac{n(\beta_0\varepsilon_0+\alpha_0\varepsilon_1)}{(\alpha_0+\beta_0)(\alpha_0+\beta_0+\mathrm{n})^2} + \\ + \frac{n^{[2-]}\left(\beta_0^{[2+]}\varepsilon_0^2 - 2\alpha_0\beta_0\varepsilon_0\varepsilon_1 + \alpha_0^{[2+]}\varepsilon_1^2\right)}{(\alpha_0+\beta_0)^{[2+]}(\alpha_0+\beta_0+n)^2}. \tag{46}$$

If the true values of the parameters α, β are unknown and MM-estimates are used, then the following asymptotic expansion for the mean square error of the predictor (5) hold:

$$\breve{r}^2_{bayes} = \frac{\alpha_0\beta_0}{(\alpha_0+\beta_0)^{[2+]}(\alpha_0+\beta_0+n)} + \\ +2\frac{\beta_0\left(n(\beta_0+1)-\alpha_0^2-\alpha_0\beta_0\right)\varepsilon_0 + \alpha_0\left(n(\alpha_0+1)-\beta_0^2-\alpha_0\beta_0\right)\varepsilon_1}{(\alpha_0+\beta_0)^{[2+]}(\alpha_0+\beta_0+n)^2} + o(\varepsilon_0) + o(\varepsilon_1). \tag{47}$$

These equations lead us to conclusion that the classical Bayes predictor looses its optimality under distortions.

4.2 New Optimal Bayes Predictor under Distortions

As a result, a new Bayes-based predictor that takes into account possible binary data distortions and provides minimum mean square forecast error needs to be developed. Let us have some estimates $\hat{\alpha}$, $\hat{\beta}$, ε_0, ε_1 of the beta-binomial model parameters and of the distortion levels that are obtained using one of the methods from the previous section. Then, it can be shown that the probability density function, the posterior mean and the mean square forecast error of the new predictor are defined as:

$$f_p(x|s,\varepsilon_0,\varepsilon_1) = \sum\nolimits_{i=0}^{n} \omega_{si} \cdot f_{\xi_i}(x), \quad L\{\xi_i\} = B(\hat{\alpha}+i, \hat{\beta}+n-i), \tag{48}$$

$$\hat{\tilde{p}}_{bayes}(s) = E\{p|s,\varepsilon_0,\varepsilon_1\} = \sum_{i=0}^{n} \omega_{si} \cdot (\hat{\alpha}+i) / (\hat{\alpha}+\hat{\beta}+n), \tag{49}$$

$$\omega_{si} = C_n^i w_{si} B(\hat{\alpha}+i, \hat{\beta}+n-i) \Big/ \sum_{j=0}^{n} \left(C_n^j w_{sj} B(\hat{\alpha}+j, \hat{\beta}+n-j) \right), \tag{50}$$

$$\tilde{r}^2_{\min}\left(\hat{\tilde{p}}_{bayes}\right) = \frac{\alpha^{[2+]}}{(\alpha+\beta)^{[2+]}} - \sum\nolimits_{i=0}^{n} \left(\left(\sum\nolimits_{j=0}^{n} \omega_{ij} \frac{\alpha+j}{\alpha+\beta+n} \right)^2 \cdot \sum\nolimits_{j=0}^{n} w_{ij} C_n^j \frac{\alpha^{[j+]} \beta^{[(n-j)+]}}{(\alpha+\beta)^{[n+]}} \right), \tag{51}$$

where s is a distorted number of responses of the customer in the past n campaigns. Let us note that when no distortions are present, the suggested predictor is identical to the classical Bayes predictor (6).

5. COMPUTER SIMULATION

To demonstrate the efficiency of the developed robust estimation and forecasting techniques, the following computer simulation was performed. Assuming that the true parameter values of the beta distribution were $\alpha_0 = 0.5$, $\beta_0 = 9.5$, there were generated K= 10000 realizations of the beta random variable with the parameters α_0, β_0. Then, for each realization (object), random Bernoulli sample of size n=10 was generated. Every sample was distorted using the expression (6); it was assumed that $\varepsilon_0 = \varepsilon_1 \in \{0, 0.01, \ldots, 0.05\}$. For each distortion level, there were calculated estimates of the α, β parameters (using MML and MLSE), and for each object, forecast of response probability was performed using the expressions (5) (based the on MML estimates) and (49) (based on the MLSE estimates). Finally, for every distortion level, 95-percent confidential intervals (CI) of the mean square forecast error were computed for the classical and proposed predictors. The results of the computer simulation are presented in Table 5-1. As follows from the table, the proposed forecasting method together with the developed simultaneous estimation algorithm ensures much lower mean square forecast error when compared to the classical approach.

Table 5-1. Comparison of the classical and proposed predictors

Distortion level	CI bounds for classical predictor		CI bounds for proposed predictor	
ε	left	right	left	right
0.00	0.0021	0.0024	0.0021	0.0024
0.01	0.0024	0.0026	0.0023	0.0025
0.02	0.0028	0.0030	0.0024	0.0027
0.03	0.0034	0.0036	0.0026	0.0029
0.04	0.0040	0.0042	0.0026	0.0028
0.05	0.0049	0.0051	0.0027	0.0030

6. APPLICATION OF THE DEVELOPED METHODS TO MEDIAPLANNING

The developed methods of robust modelling of consumer behaviour based on the proposed distorted beta-binomial model have been implemented in the commercial software package Pinergy (Omega Software GmbH) for predicting the behaviour of TV audience. To test the efficiency of the developed algorithms, real-life data for one of the German TV stations for the period 02.01.1999-01.01.2000 was used. Using this set of data, the developed distorted beta-binomial model was compared to the well known classical beta-binomial model for forecasting the Net Reach and Gross Rating Points (GRP) for master target groups of TV audience.

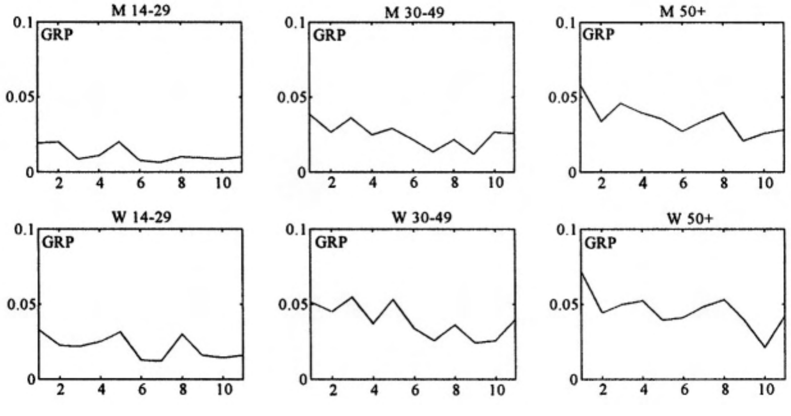


Figure 5-1. Gross Rating Points for the selected set of TV breaks, master target groups

Table 5-2. Adequacy of the beta-binomial model (the classical MM)

Target group	M 14-29	M 30-49	M 50+	W 14-29	W 30-49	W 50+
p-value	0.41	0.96	0.01	0.82	0.10	0.05
χ^2 – statistics	9.2970	3.1358	21.7483	5.1632	14.5561	17.0657
Parameter α	0.17	0.16	0.17	0.18	0.21	0.16
Parameter β	13.87	3.14	4.56	8.33	5.27	3.25

Table 5-3. Adequacy of the beta-binomial model (the classical MML)

Target group	M 14-29	M 30-49	M 50+	W 14-29	W 30-49	W 50+
p-value	0.45	0.97	0.02	0.80	0.10	0.05
χ^2 – statistics	8.8783	2.8660	20.0282	5.4255	14.6119	16.8589
Parameter α	0.17	0.17	0.19	0.19	0.24	0.17
Parameter β	13.53	6.55	5.08	8.81	5.78	3.47

Table 5-4. Adequacy of the distorted beta-binomial model (the developed method of simultaneous estimation of the parameters and the distortion levels based on the MM)

Target group	M 14-29	M 30-49	M 50+	W 14-29	W 30-49	W 50+
p-value	0.28	0.99	0.32	0.89	0.84	0.14
χ^2 – statistics	10.8993	0.9956	10.4146	4.3460	4.9850	13.4565
Parameter α	0.09	0.13	0.14	0.13	0.15	0.15
Parameter β	9.57	5.58	3.88	7.07	4.00	3.14
Distortion level ε_0	0.003	0.002	0.003	0.003	0.006	0.001
Distortion level ε_1	0.060	0.000	0.042	0.000	0.06	0.004

Table 5-5. Adequacy of the distorted beta-binomial model (the developed method of simultaneous estimation of the parameters and the distortion levels based on the MML)

Target group	M 14-29	M 30-49	M 50+	W 14-29	W 30-49	W 50+
p-value	0.63	0.98	0.77	0.88	0.83	0.50
χ^2 – statistics	7.1145	2.4552	5.6728	4.3983	5.0365	8.3400
Parameter α	0.11	0.15	0.10	0.15	0.15	0.11
Parameter β	9.12	6.39	3.14	7.51	4.31	2.36
Distortion level ε_0	0.001	0.002	0.007	0.002	0.006	0.005
Distortion level ε_1	0.048	0.015	0.068	0.008	0.020	0.111

Since in practice the problem of forecasting the behaviour of the TV audience is usually separated into a number of sub-problems for similar TV breaks [14], 11 TV commercial breaks that correspond to the fixed program (world news), week day (Saturday) and day time (prime time) were selected. The GRPs for these breaks are shown in Figure 5-1; since GRPs deviation from break to break is quite low, the assumption of the (distorted) beta-binomial model can be made.

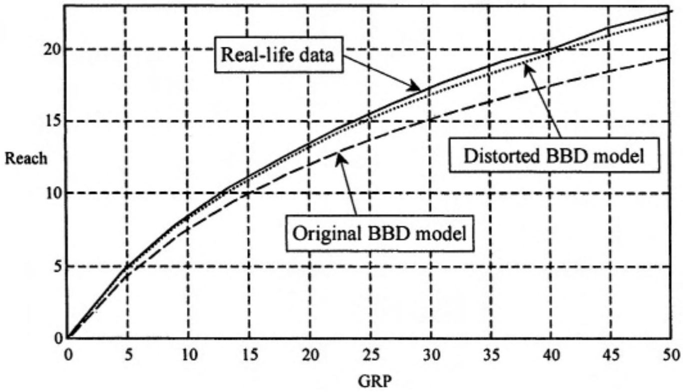


Figure 5-2. Reach/GRP curves: real data (solid line), classical beta-binomial model (dashed line), the proposed distorted beta-binomial model (dotted line)

Tables 5-2, 5-3, 5-4, 5-5 compare the adequacy of the classical beta-binomial model and the distorted beta-binomial model for different estimation methods using Pearson's χ^2 goodness-of-fit test. As follows from the tables, the proposed distorted beta-binomial model together with the developed estimation algorithms significantly increases the modeling accuracy. For example, for the target group "Males that are 50 years of age and older" (M 50+) the adequacy of the classical beta-binomial model is very low and is characterized by the p-values 0.01 for the method of moments and 0.02 for the method of maximum likelihood. In contrast, the adequacy of the proposed distorted beta-binomial model is very high and is characterized by the p-values 0.32 for the simultaneous method of estimation based on the method of moments and 0.77 for the simultaneous method of estimation based on the method of maximum likelihood.

In practice, media-planners evaluate the adequacy of the customer's respond model using Reach/GRP curves [14]. Figure 5-2 compares Reach/GRP curves for the classical beta-binomial model and for the proposed distorted beta-binomial model with the real data. As follows from the figure, the proposed distorted beta-binomial model together with the developed estimation algorithms ensures much more accurate approximation of the Reach/GRP function then the classical beta-binomial model.

7. CONCLUSION

In this chapter, a new model for describing the probability of customer respond in database marketing is proposed. New robust estimation and forecasting algorithms are developed for this model. The efficiency of the proposed methods is verified via computer simulation and tests on real data from mediaplanning.

Further work will deal with applying robust logistic regression methods to the problem of forecasting the response probabilities.

ACKNOWLEDGEMENTS

This research was partly supported by Belarussian State University grant for young scientists (533/30) and Omega Software GmbH company.

REFERENCES

1. Buchanan B.S. and Morrison D.G., 1988. Stochastic Modeling of List Falloffs with Implications for Repeat Mailings. *Journal of Direct Marketing, 2,* 7-14.
2. Collet D., 2002. *Modeling binary data,* Champton & Hall/CRC, London.
3. Colombo R. and W. Jiang, 1998. A stochastic RFM Model. *Journal of Interactive Marketing,* 13, 2-12.
4. Copas J.B., 1988. Binary regression models for contaminated data. *Journal of Royal Statistical Society,* 50(B), 225-265.
5. DeSarbo W. and Ramaswamy V., 1994. CRISP: Customer Response Based Iterative Segmentation Procedures for Response Modeling in Direct Marketing. *Journal of Direct Marketing,* 8, 7-20.
6. Diggle P.J., Liang K.-Y. and Zeger S.L., 2002. *Analysis of Longitudinal Data,* Clarendon Press, Oxford.
7. Haughton D. and Oulabi S., 1993. Direct Marketing Modeling with CART and CHAID. *Journal of Direct Marketing,* 7, 16-26.
8. Hampel F.R., Rousseeuw P.J., Ronchetti E.M. and Stahel W.A., 1986. *Robust Statistics,* John Wiley and Sons, N. Y.
9. Huber P.J., 1981. *Robust Statistics,* Wiley , N. Y.
10. Johnson N.L., Kotz S. and Kemp A.W., 1996. *Univariate Discrete Distributions,* Wiley-Interscience, N.Y.
11. Kharin Yu., 1996. *Robustness in Statistical Pattern Recognition,* Kluwer Academic Publishers, Dordrecht.
12. Massy W.F., Montgomery D.B. and Morrison D.G., 1970. *Stochastic models of Buying Behavior,* The MIT Press, Cambridge, Mass..
13. Leahy K., 1992. Beyond Prediction. *Journal of Direct Marketing,* 6, 9-16.
14. Rust R., 1986. *Advertising Media Models: A Practical Guide,* Lexington Books, Lexington, Mass.
15. Shepherd D., 1990. *The New Direct Marketing,* Business One Irwin, Homewood, IL.
16. Tripathi R.C., Gupta R.C. and Gurland J., 1994. Estimation of parameters in the beta binomial model. *Ann. Inst. Statist. Math.,* 46, 317-331.
17. Zahavi J. and Levin N., 1995. Issues and Problems in Applying Neural Computing to Target Marketing. *Journal of Direct Marketing, 9,* 33-45.

Chapter 6

SIZING, CYCLE TIME AND PLANT CONTROL USING DIOID ALGEBRA

Said Amari, Isabel Demongodin, Jean-Jacques Loiseau

Abstract: Using an industrial process from the car sector, we show how dioid algebra may be used for the performance evaluation, sizing, and control of this discrete-event dynamic system. Based on a Petri net model as an event graph, max-plus algebra and min-plus algebra permit to write linear equations of the behavior. From this formalism, the cycle time is determined and an optimal sizing is characterized for a required cyclic behavior. Finally, a strict temporal constraint the system is subject to is reformulated in terms of inequalities that the (min, +) system should satisfy, and a control law is designed so that the controlled system satisfies the constraint.

Key words: dioid algebra, Petri net, sizing, cycle time, control.

1. INTRODUCTION

For efficient design and operation techniques of manufacturing systems, it is well known that methods and tools are needed to model complex material and control flows, to analyze behavior and interactions of manufacturing resources and to predict the performance measures such as productivity, cycle times and work-in-process. Thanks to significant advances over the last decade, discrete-event system techniques have gained maturity and provide a wide spectrum of tools for solving problems encountered in design, supervisory control and performance evaluation of manufacturing systems. Petri nets, queuing networks, dioid algebra, perturbation analysis, formal language theory, state-charts have, among many others, proven techniques for modeling, specification and analysis of complex discrete-event dynamic systems (DEDS).

The problem tackled here is the performance evaluation, the optimal sizing and the control of a manufacturing plant from the sector of the car industry. The studied process can be embedded in the discrete-event dynamic system class. Among various techniques of analysis and control [1], a theory of linear algebra for DEDS also known as dioid theory has been recently developed [2]. More precisely, as long as the DEDS has no conflict, it can be modeled by an event graph or marked graph, i.e. by a Petri net in which each place has exactly one input and one output transitions. Such a graph is called a Timed Event Graph (TEG) when a sojourn time is associated to each place, or by duality, when a delay is associated to each transition. Hence, the timed behavior of such a TEG is represented by the dates at which the transitions of a TEG are fired for the k^{th} time or equivalently, by the number of firings of transitions at time t. The former description, as daters in the events domain, leads to linear equations in the so-called max-plus algebra, while in the latter, as counters in the time domain, equations are linear in the min-plus algebra. In both cases, there exists a spectral theory that leads to one of the main results in that frameworks concerning performance evaluation of DEDS. The cycle time, or the throughput, is the solution of an eigenvalue problem in the considered algebra. Beside these performance evaluation interests, control problems have been addressed in the context of DEDS using this algebra. In this chapter, the first step concerns the cycle time of the industrial plant using the max-plus algebra. Next, for a required cycle time, the sizing and the control of the plant are determined using the min-plus algebra.

This chapter is organized as follows. Backgrounds concerning the dioid algebra are recalled in Section 2. A brief description of the industrial plant is given in Section 3. A Timed Event Graph model of the plant is provided in Section 4, and linear equations associated to the system in the max-plus dioid algebra are derived in Section 5. On the basis of that model, the cycle time is determined in the next section for a given configuration, that is a given number of available pallets. The sizing problem is tackled in Section 7, where the number of free pallets is optimized for a required cycle time. Finally a control problem is addressed and solved in Section 8.

2. DIOID ALGEBRA

Definition 1 *(Dioid algebra).*

(i) An abelian monoid is a set D endowed with an internal law $\oplus$ that is associative, commutative, and admits a neutral element denoted ε.

(ii) An abelian semiring is a monoid endowed with a second operation $\otimes$, also associative, distributive with respect to the first law $\oplus$, admitting a neutral element e, and so that the neutral element of the first law is absorbing for the second law $\otimes$ (i.e. $\varepsilon \otimes a = a \otimes \varepsilon = \varepsilon, \forall a \in D$).

(iii) A dioid is a semiring with an idempotent first law (i.e. $a \oplus a = a, \forall a \in D$).

One says that the dioid is commutative provided that the law $\otimes$ is commutative.

Example 1 *(Max-plus algebra).* $\overline{\mathfrak{R}}_{\max} = \{\mathfrak{R} \cup \{-\infty\} \cup \{+\infty\}, \max, +\}$ is a commutative dioid, with zero element ε equal to $-\infty$, and the unit element e equal to 0. We adopt the usual notation, so that the symbol $\oplus$ stands for the max operation, and $\otimes$ stands for the addition. Notice that $\varepsilon \otimes +\infty = (-\infty) + (+\infty) = \varepsilon = -\infty$ in $\overline{\mathfrak{R}}_{\max}$.

Example 2 *(Min-plus algebra).* $\overline{\mathfrak{R}}_{\min} = \{\mathfrak{R} \cup \{-\infty\} \cup \{+\infty\}, \min, +\}$ is also a commutative dioid, for which ε equals to $+\infty$ and e equals to 0. We shall denote $\oplus$ ' the min operation in the sequel, and the symbol $\otimes$ ' will stand for the addition. Notice that $\varepsilon \otimes -\infty = (+\infty) + (-\infty) = \varepsilon = +\infty$ in $\overline{\mathfrak{R}}_{\min}$.

Example 3 *(Matrix dioid).* Let $(D, \oplus, \otimes)$ be a given dioid, and denote $D^{n \times n}$ the set of square $n \times n$ matrices with entries over D. The sum and the product over D extend as usually over $D^{n \times n}$ as follows:

$$(A \oplus B)_{ij} = A_{ij} \oplus B_{ij} \text{ and } (A \otimes B)_{ij} = \overset{n}{\underset{k=1}{\oplus}} A_{ik} \oplus B_{kj}.$$

One can see that $(D^{n \times n}, \oplus, \otimes)$ is a dioid. The neutral elements for the law $\oplus$ is the matrix the entries of which equal ε, the neutral element for the law $\otimes$ is the matrix the entries of which equal e on the diagonal and ε outside. Notice that the products of matrices in $\overline{\mathfrak{R}}_{\max}$ and in $\overline{\mathfrak{R}}_{\min}$ are not equal, (and do not equal the usual sum of matrices.)

Definition 2 *(Trace).* The trace of a square matrix $A \in D^{n \times n}$ is the sum of its diagonal entries, denoted:

$$tr(A) = \overset{n}{\underset{i=1}{\oplus}} A_{ii}.$$

In the max-plus algebra, we shall also use the notation:

$$\left(tr(A^k)\right)^{\frac{1}{k}} = \frac{1}{k}\left(\overset{n}{\underset{i=1}{\oplus}} \left(A^k\right)_{ii}\right), \text{ where } A^k = \underbrace{A \otimes \cdots \otimes A}_{k \text{ times}}.$$

These concepts are useful to model Timed Event Graphs with sojourn delays associated to places. A TEG is an ordinary Petri net where each place has a single input transition and a single output transition. We denote p_{ij} the place from transition t_j to transition t_i. We assume that n is the number of transitions which are upstream and downstream transitions of places. The number of sink transitions, i.e. without output places, is equal to s. The number of transitions called source, having no downstream place, is m.

Delays or holding times are associated to places of the TEG. Thus the graph is timed on places. The max-plus approach [2] allows us to model with a system of linear inequations the dynamic behavior of such a TEG.

$$\begin{cases} x(k+1) \geq A \otimes x(k) \oplus B \otimes u(k+1), \\ y(k) \geq C \otimes x(k), \end{cases} \tag{1}$$

where the components of the vector $x(k)$ are the firing times of the n transitions t_i for the k^{th} occurrence, the components of $u(k)$ and of $y(k)$ are respectively the firing dates of the source and of the sink transitions. A, B and C are respectively matrices of size $n \times n$, $n \times m$ and $s \times n$.

3. PLANT DESCRIPTION

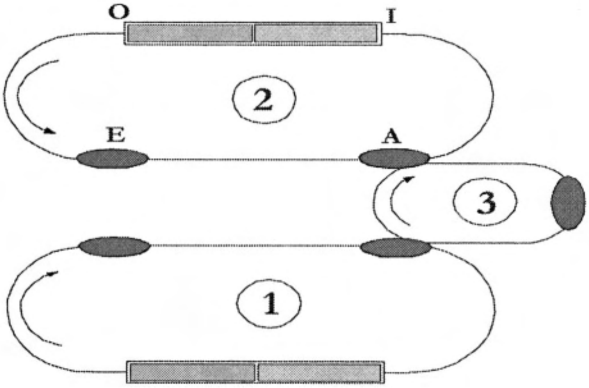


Figure 6-1. The plant

The process we study here (see reference [3] for more details) is composed of three conveyors belt connected by loops (Figure 6-1). The parts are made on an extruding machine on loop 3. Loops 1 and 2 are both similar one to each other; they are dedicated to a thermal processing of the parts. Loop 3 processes parts that are conveyed on pallets to one of the other loops.

On loop 1, respectively on loop 2, pallets are devoted for a certain type of parts coming from loop 3. Hence, synchronization is needed between these three transfer elements. The machine on loop 3 is very flexible and therefore can provide any type of part needed on loop 1 or loop 2. The main problem is to achieve the thermal treatment on loop 1 or loop 2 without major failures.

Assume loop 2 is under study (identical process for loop 1) (see Figure 6-2). Parts arrive (from loop 3) at point A and an operator fixes them to a pallet. The pallets are then released on a conveyor belt that leads them to point I. Here they enter inside the furnace. This element is a channel divided into two sections. Inside the former section parts are heated and they are next cooled down inside the latter. We can remark that there is no buffer between these two sections. Once pallets come outside the furnace (point O), they are transferred to a second operator who removes parts from the pallets. Thus, parts are taken away at point E according to the external resources. Finally, the free pallets are released and transfer to point A.

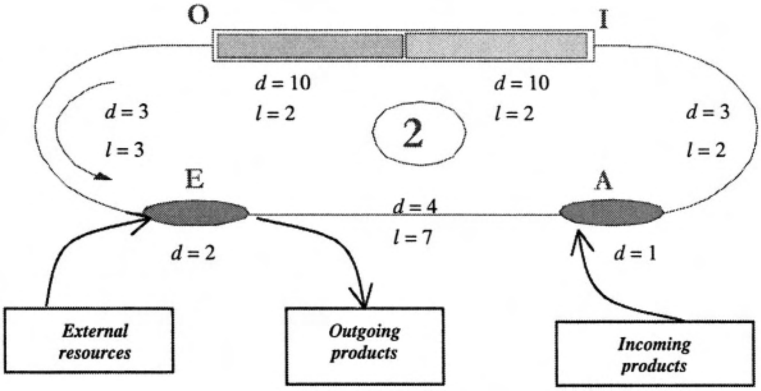


Figure 6-2. Loop 2

The constraint for the thermal processing is that no overrun on the heating time is permitted. If so, all parts are rejected and the whole process has to be restarted. The problem is that from time to time, parts are not taken away immediately and those perturbations can affect the production achievement if the sizing, in number of pallets involved, is not correctly designed.

4. TIMED EVENT GRAPH OF THE LOOP

Let us consider loop 2. The capacity of conveyor E-A is equal to 7 pallets, while 5 pallets are free on this conveyor. This physical process is modeled thanks to a TEG with sojourn delays associated to places. The durations of operations (noted *d* in Figure 6-2) are assumed to be identical for all the parts and are reported beside places on the Petri net model (see Figure 6-3). For instance, the transfer time from point E to point A is considered to be equal to four time units and is represented by place p_{17} (i.e. place from transition t_7 to transition t_1). Tokens represent the resources of the plant: the pallets, the operators or the conveyor capacities (noted l in Figure 6-2). Transition t_1 models the beginning of the fixing operation and transition t_2 the end of that operation. Only one token is available for this firing transition, meaning that the operator can treat one part at a time. The delay spent for this task is equal to one time unit. The beginning of the thermal process, point I, is modeled by the firing of transition t_3.

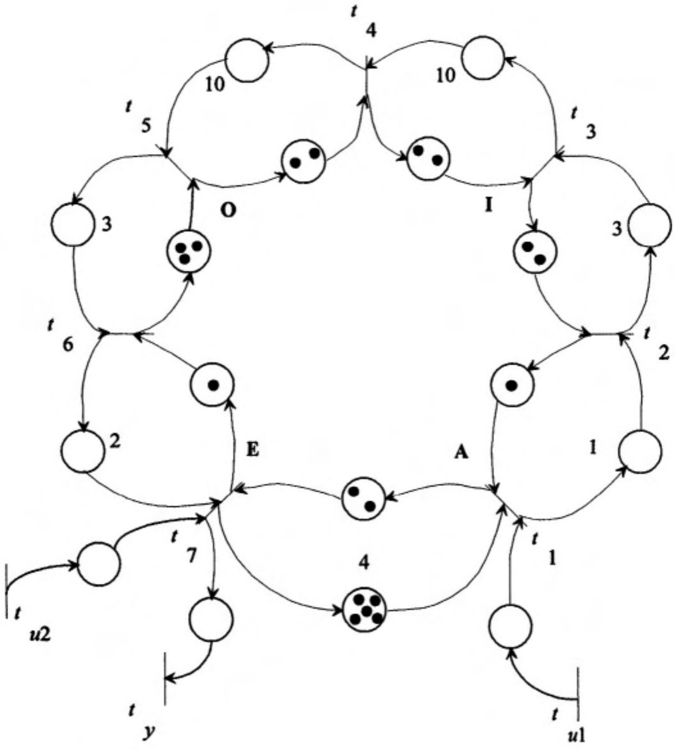


Figure 6-3. Model of the plant– loop 2

Transition t_y represents the departure of an achieved part, t_{u2} models the necessity of a resource to carry the terminated part and t_{u1} models parts arrivals from loop 3. Figure 6-3 shows the process in an initial state where the five pallets involving through the system are available. The maximal capacity of the conveyor from point E to point A is supposed to be equal to seven pallets.

5. DIOID MODEL OF THE LOOP

The construction of matrix *A,* see (1), can be done in several different ways. Olsder et al. have proposed in [4] a method to built max-plus models for large scale systems. For relatively small scale systems, like in the case of this study, one can rather apply the method proposed by Mairesse [5]. That method consists in an addition of some places and transitions in order to obtain a TEG with places containing either zero or one token. A null delay is associated to those additional places. One can see in Figure 6-4 the resulting TEG, applying that method to the model of Figure 6-3.

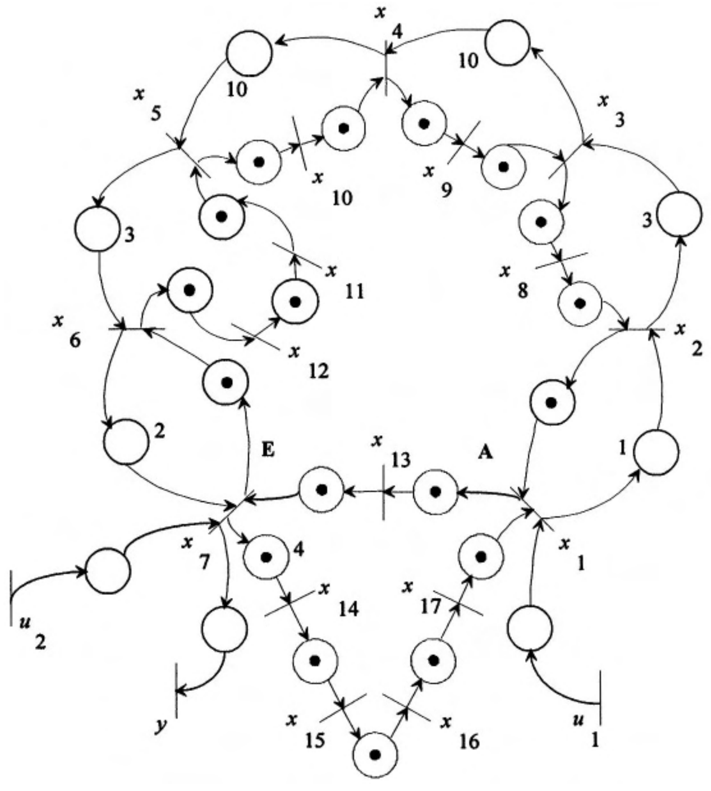


Figure 6-4. Resulting TEG for loop 2

The TEG dynamic behavior can therefore be modeled by the following inequations (symbol ⊗ is omitted for the seek of readability):

$$\begin{cases} x(k+1) \geq A_0 x(k+1) \oplus A_1 x(k) \oplus B_0 u(k+1), \\ y(k) \geq Cx(k). \end{cases} \tag{2}$$

Then, matrices A_0, A_1 and B_0 are built. A_{0ij} is the delay associated to the empty place that links transition t_j to transition t_i. A_{1ij} is the delay associated to the place containing one token that links transition t_j to transition t_i. B_{0ij} is the delay associated to the empty place that links transition t_{uj} to transition t_i. If there is no place linking transition t_j to transition t_i (resp. transition t_{uj} to transition t_i) the entry is ε. If the delay is zero, it is noted e. Matrices A_0, A_1, B_0 and C associated to the TEG of Figure 6-3 are given bellow, where ε is replaced by a dot for the seek of readability.

$$A_0 = \begin{pmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & 3 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & 10 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & 10 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & 3 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & 2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot
\end{pmatrix}
\quad
A_1 = \begin{pmatrix}
\cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 4 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e
\end{pmatrix}$$

$$B_0 = \begin{pmatrix}
e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot
\end{pmatrix}^T$$

$$C = \begin{pmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot
\end{pmatrix}$$

For TEG under maximal speed assumption, system of inequations (2) turns into a system of equations. Furthermore one can derive the system of explicit equations (3):

$$\begin{cases} x(k+1) = Ax(k) \oplus Bu(k+1), \\ y(k) = Cx(k), \end{cases} \tag{3}$$

with matrices $A = A_0^* A_1$ and $B = A_0^* B_0$, where operator * stands for the Kleene star operator: $a^* = e \oplus a \oplus a^2 \oplus \ldots$.

$$A_0^* = \begin{pmatrix}
e & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
1 & e & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
4 & 3 & e & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
14 & 13 & 10 & e & . & . & . & . & . & . & . & . & . & . & . & . & . \\
24 & 23 & 20 & 10 & e & . & . & . & . & . & . & . & . & . & . & . & . \\
27 & 26 & 23 & 13 & 3 & e & . & . & . & . & . & . & . & . & . & . & . \\
29 & 28 & 25 & 15 & 5 & 2 & e & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & e & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & e & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & e & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & e & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & e & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & e & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & e & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & e & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & e & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & e
\end{pmatrix}$$

$$A = \begin{pmatrix}
. & e & . & . & . & . & . & . & . & . & . & . & . & . & . & . & e. \\
. & 1 & . & . & . & . & . & e & . & . & . & . & . & . & . & . & 1 \\
. & 4 & . & . & . & . & . & 3 & e & . & . & . & . & . & . & . & 4 \\
. & 14 & . & . & . & . & . & 13 & 10 & e. & . & . & . & . & . & . & 14 \\
. & 24 & . & . & . & . & . & 23 & 20. & 10 & e & . & . & . & . & . & 24 \\
. & 27 & . & . & . & . & e & 26 & 23 & 13 & 3 & . & . & . & . & . & 27 \\
. & 29 & . & . & . & . & 2 & 28 & 25 & 15 & 5 & . & e & . & . & . & 29 \\
. & . & e. & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & e & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & e & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & e & . & . & . & . & . & . \\
. & . & . & . & . & e & . & . & . & . & . & . & . & . & . & . & . \\
e & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & 4 & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & e & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & e & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & e
\end{pmatrix}$$

$$B = \begin{pmatrix}
e & 1 & 4 & 14 & 24 & 27 & 29 & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & e & . & . & . & . & . & . & . & . & . & .
\end{pmatrix}^T$$

6. CYCLE TIME OF THE AUTONOMOUS LOOP

The throughput is maximal if inputs do not constrain the evolution, i.e. if parts are always available at point A and transportation utilities always present at point E. This is modeled by $u_i(k) = \varepsilon$. Then, the system is said to be autonomous, without source transitions, and can be modeled by equation (4).

$$x(k+1) = Ax(k)\,. \tag{4}$$

As long as its graph is strongly connected, or equivalently, matrix A is irreducible, after a finite transient behavior, the system reaches a periodic regime. Therefore, one can write:

$$\exists K, \forall k \geq K, x(k+c) = \lambda^c x(k), \tag{5}$$

where λ refers to equation (6):

$$Ax = \lambda x, \tag{6}$$

and c is the cyclicity of matrix A. In other words, c parts are achieved every $c \times \lambda$ time units. The eigenvalue λ is unique for a strongly connected event graph and the throughput is given directly as $1/\lambda$. This eigenvalue is computed according to the algorithm of Cochet-Terrasson et al. [6].

$$\lambda = \bigoplus_{k=1}^{n} \left(trA^k\right)^{\frac{1}{k}}. \tag{7}$$

Applying this algorithm to the model of the plant, $\lambda = 33/5 = 6.6$ time units.

7. SIZING FOR A REQUIRED CYCLE TIME

Now, the capacity of the conveyor from point E to point A is not fixed (i.e. place p_{71} is empty in Figure 6-3). The sizing problem is to determine the minimal number of pallets needed to reach the cycle time of 5 units. This required cycle time corresponds to the bottleneck element, i.e. the furnace, which has a production cycle at 5 time units. In other terms, the minimal initial marking of place p_{17} has to be determined. A method to solve this problem is presented by Gaubert in [7]. After having chosen a desired periodic throughput $\overline{\varphi}$, one can calculate in the min-plus algebra, the initial minimal marking q. It is shown that the throughput constraint $\varphi(q) \geq \overline{\varphi}$ is equivalent to the existence of a finite subeigenvector of $\overline{\varphi}$. We denote by T_{ij} the holding time of place p_{ij} and by N_{ij} its initial marking. If place p_{ij} does not exist, by convention, $T_{ij} = -\infty$ and $N_{ij} = +\infty$. For the TEG of Figure 6-3, with an initial marking $q_{71} = 0$ and q_{17} unknown, these matrix are defined as follow, where the dot replaces the infinite values:

$$N(q) = \begin{bmatrix} . & 1 & . & . & . & . & q_{17} \\ e & . & 2 & . & . & . & . \\ . & e & . & 2 & . & . & . \\ . & . & e & . & 2 & . & . \\ . & . & . & e & . & 3 & . \\ . & . & . & . & e & . & 1 \\ e & . & . & . & . & e & . \end{bmatrix} \text{ and } T = \begin{bmatrix} . & e & . & . & . & . & 4 \\ 1 & . & e & . & . & . & . \\ . & 3 & . & e & . & . & . \\ . & . & 10 & . & e & . & . \\ . & . & . & 10 & . & e & . \\ . & . & . & . & 3 & . & e \\ e & . & . & . & . & 2 & . \end{bmatrix}$$

From the method of Gaubert [7], the following assertions are equivalent:

i) $\varphi(q) \geq \overline{\varphi}$.

ii) $\rho(C(q)) \geq 0$, (with $\rho(C(q)) = \bigoplus_{k=1}^{n}{}' \left(trC(q)^k\right)^{\frac{1}{k}}$).

where matrix *C(q)* is defined by: $C_{ij}(q) = N_{ij}(q) - \overline{\varphi} T_{ij}$.

iii) There exists a vector $\alpha \in \Re^n$ such that $\forall i$, $\min_j (C_{ij}(q) + \alpha_j) \geq \alpha_i$.

We can notice that $C_{ij}(q) = +\infty$ if either $N_{ij}(q) = +\infty$ or $T_{ij}(q) = +\infty$. This ensures the coherence of the notation ‘dot’.

For $\overline{\varphi} = 1/5$, $C(q) = N(q) - \frac{1}{5}T$, and thus:

$$C(q) = \begin{bmatrix} . & 1 & . & . & . & . & (q_{17} - \frac{4}{5}) \\ -\frac{1}{5} & . & 2 & . & . & . & . \\ . & -\frac{3}{5} & . & 2 & . & . & . \\ . & . & -\frac{10}{5} & . & 2 & . & . \\ . & . & . & -\frac{10}{5} & . & 3 & . \\ . & . & . & . & -\frac{3}{5} & . & 1 \\ e & . & . & . & . & -\frac{2}{5} & . \end{bmatrix}$$

Applying assertion ii) with $\overline{\varphi} = 1/5$, a minimal initial marking of place p_{17} is equal to 7, as $q_{17} \geq 33/5$. Finally, a minimum of 7 free pallets guarantees to the process a cycle time of 5 units.

8. CONTROL UNDER STRICT TIME CONSTRAINT

The production unit includes a critical section, which is the warm part of the furnace. The parts should not stay more than 10 time units in this section. Hence the production system is submitted to a strict timed constraint, which is the required sojourn time in the oven. The non respect of the constraint can occur, for instance, because of the lack of transportation resource (point E of Figure 6-2) during a certain period of time which leads to a blocking of pieces in the furnace. For the normal behavior of the process, the finite product is taken away and the pallets are recycled at the level of point E. In case of a problem at this point, the parts pile up downstream the working station, which may lead to the blocking of parts in the furnace and the cessation of the production.

8.1 Min-plus model: explicit equation

To each transition t_i we associate the time function $\theta_i(t)$, which is the number of firings of transition t_i at time t. For the TEG of Figure 6-3, $U_l(t)$, which is the number of firings of transition t_{ul} at time t, can be seen as a control, which permits to retain the parts arriving from the central loop 3, and postpone their arrival in the production process, if necessary. Similarly, $U_2(t)$ can be seen as a disturbance, i.e. the number of firings of transition t_{u2} at time t. In the sequel, $U(t)$ denotes the vector with components $U_l(t)$ and $U_2(t)$. The min-plus equations of the production unit are deduced from the event graph of Figure 6-3. They read like follows:

$$\begin{aligned}\theta(t) = \; & A_0^{'*} A_1^{'} \theta(t-1) \oplus' A_0^{'*} A_2^{'} \theta(t-2) \oplus' A_0^{'*} A_3^{'} \theta(t-3) \oplus' A_0^{'*} A_4^{'} \theta(t-4) \\ & \oplus' A_0^{'*} A_{10}^{'} \theta(t-10) \; \oplus' A_0^{'*} B' U(t),\end{aligned} \tag{8}$$

where:

$$A_1^{'} = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ e & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \quad A_2^{'} = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & e & \cdot \end{bmatrix} \quad A_3^{'} = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & e & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & e & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix}$$

$$A_4^{'}=\begin{bmatrix} . & . & . & . & . & . & 7 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \end{bmatrix} \quad A_{10}^{'}=\begin{bmatrix} . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & e & . & . & . & . \\ . & . & . & e & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \end{bmatrix} \quad B^{'}=\begin{bmatrix} e & . \\ . & . \\ . & . \\ . & . \\ . & . \\ . & . \\ . & e \end{bmatrix}$$

$$A_0^{'}=\begin{bmatrix} . & 1 & . & . & . & . & . \\ . & . & 2 & . & . & . & . \\ . & . & . & 2 & . & . & . \\ . & . & . & . & 2 & . & . \\ . & . & . & . & . & 3 & . \\ . & . & . & . & . & . & 1 \\ e & . & . & . & . & . & . \end{bmatrix} \text{ so that } A_0^{'*}=\begin{bmatrix} e & 1 & 3 & 5 & 7 & 10 & 11 \\ 10 & e & 2 & 4 & 6 & 9 & 10 \\ 8 & 9 & e & 2 & 4 & 7 & 8 \\ 6 & 7 & 9 & e & 2 & 5 & 6 \\ 4 & 5 & 7 & 9 & e & 3 & 4 \\ 1 & 2 & 4 & 6 & 8 & e & 1 \\ e & 1 & 3 & 5 & 7 & 10 & e \end{bmatrix}$$

8.2 Time constraint

The strict timed constraint holds at the level of the place p_{43} (linking transition t_3 to transition t_4) which represents the warm part of the furnace. The duration of the thermal treatment is fixed and equals to 10 time units. This constraint can be expressed in terms of the variables $\theta_i(t)$, as follows:

$$\theta_3(t)-\theta_4(t+10)=0, \forall t \geq 10. \tag{9}$$

The control problem we face consists in synthesizing a control $U_l(t)$, knowing $\theta_i(t)$ and $U_2(t)$, which guarantees the respect of the constraint (9).

Replacing the matrices in the explicit equation (8), one obtains:

$$\begin{aligned} &\theta_4(t)=7\theta_1(t-1)\oplus^{'}6\theta_6(t-2)\oplus^{'}9\theta_2(t-3)\oplus^{'}5\theta_5(t-3) \\ &\oplus^{'}13\theta_7(t-4)\oplus^{'}\theta_3(t-10)\oplus^{'}2\theta_4(t-10)\oplus^{'}6U_1(t)\oplus^{'}6U_2(t) \end{aligned} \tag{10}$$

from which the following equivalence is deduced:

$$\theta_4(t+10)=\theta_3(t) \qquad \Leftrightarrow \qquad \begin{cases} 2\,\theta_4(t)\geq\theta_3(t) & (a) \\ 5\,\theta_5(t+7)\geq\theta_3(t) & (b) \\ 6\theta_6(t+8)\geq\theta_3(t) & (c) \\ 13\theta_7(t+6)\geq\theta_3(t) & (e) \\ 9\theta_2(t+7)\geq\theta_3(t) & (f) \\ 7\theta_1(t+9)\geq\theta_3(t) & (g) \\ 6U_1(t+10)\geq\theta_3(t) & (h) \\ 6U_2(t+10)\geq\theta_3(t) & (i) \end{cases} \qquad (11)$$

One can observe that all these inequalities, except (11-*i*), readily follow from system (8) of explicit equations. The conclusion is that, provided that (8) holds, the time constraint (9) is satisfied if and only if (11-*i*) is satisfied.

8.3 Control synthesis

Remark that, on the one hand, $U_2(t+10)\geq\theta_7(t+10)$, which leads to $U_2(t+10)\geq\theta_7(t)$, since the function $\theta_7(t)$ is nondecreasing, and that, on the other hand, $U_1(t)\geq\theta_1(t)\geq\theta_2(t)\geq\theta_3(t)$ as system (8) is satisfied. It readily follows that the control law

$$U_1(t)=6\,\theta_7(t) \qquad (12)$$

implies that the condition (11-*i*) is satisfied, and consequently that the strict timed constraint is fulfilled.

The control law (12) can be interpreted as a feedback, determining at each instant *t* the control $U_1(t)$ in terms of the state $\theta_7(t)$ of the system. This control law is actually a Kanban, which lets a maximum of 6 pallets enter the production loop between point A and point E at a time, to avoid any congestion that may be caused by a failure in the working place E. Repeating for the new configuration the computation of the time cycle as in Section 6, one finds that the cycle time value is unchanged and equal to 5 time units.

9. CONCLUSION

We have applied to an industrial process in the car industry, a method for dealing with performance evaluation and sizing based on dioid algebra. First, the system has been specified and modeled as a timed discrete event graph, i.e. a Petri net structure without conflict situation. Next, such model has been dealt with max-plus algebra that provides, through semi-linear analysis type methods, an easy way to evaluate the cycle time. Using min-plus algebra, the minimal number of resources required has been determined for an imposed cycle time, and finally a control law has been designed for ensuring the respect of a strict time constraint which exists at the level of a furnace. The control law is optimal since it does not change the requiered cycle time and ensures the robustness of the production.

REFERENCES

1. Cassandras C.G. and S. Lafortune. 1999. *Introduction to discrete event systems.* Kluwer Academic.
2. Baccelli F., G. Cohen, G. Olsder, et J. Quadrat. 1992. *Synchronization and Linearity: An algebrafor Discrete Event Systems.* Willey.
3. Martinez C. and P. Castagna. 2003. Sizing of an industrial plant using tight time constraints using complementary approaches: (max, +) algebra and computer simulation. *Simulation Modelling Practice and Theory* 11, 75-88.
4. Olsder G.J., Subiono, M.Mc Guettrick. 1998. On large scale max-plus algebra model in railway systems. *Actes de la 26^{ième} école de printemps d'informatique théorique: algebre max-plus et applications en informatique et automatique,* INRIA, LIAFA, IRCyN, France, 177-192.
5. Mairesse J. 1998. Petri nets, (max,+) algebra and scheduling. *Actes de la 26^{ième} école de printemps d'informatique théorique: algebre max-plus et applications en informatique et automatique,* INRIA, LIAFA, IRCyN, France, 329-357.
6. Cochet-Terrasson J., G.Cohen, S.Gaubert, M.Mc Gettrick, J.P.Quadrat. 1998. Numerical computation of spectral elements in max-plus algebra. *IFAC conference on system structure and control,* France, 699-706.
7. Gaubert S. 1995. Resource optimization and (min,+) spectral theory. *IEEE trans. on automatic control* 40(11), 1931 -1934.

This page intentionally left blank

Chapter 7

CONCURRENT PROCESSES FLOW PROTOTYPING

Zbigniew Banaszak, Michał Polak

Abstract: The problem of controlling concurrent processes that compete for access to shared resources is considered. The processes are controlled by a pair (an initial state, a set of dispatching rules) that guarantees steady cyclic flows. A proposal of an automated prototyping procedure of a system is considered to aid determining desired system flows for assumed factors such as system capacity or the value of rate of resource utilisation.

Key words: repetitive processes, distributed control, deadlock avoidance, priority rule, resource utilisation.

1. INTRODUCTION

Process occurring in various fields of supply chain application such as: data flow in computer networks, traffic flow, product flows in factories, etc., have one feature in common - they need to be executed concurrently. In a situation, when process routes go through different, mutually excluding subsets of system resources, all the component processes are executing parallel and no process has any influence on another one. The processes' executions are limited only by the throughput of resources. However, in case when the routes of individual processes go through common (shared) system resources, the process flow depends on two factors: the first one is the shared resource capacities and the other one the rules controlling the access to such kind of resources. The resource capacity can be understood as the ability to independent, simultaneous handling of one or more processes (depending on the resource capacity).

It is worth to note that assuming a rule to synchronise the access to the resources by the processes, requires, in general, to choose an initial, acceptable allocation of the processes to the resources. The initial allocation is required, because some arbitrary process allocations, along with the rules limiting access to the resources, may lead to a deadlock.

Previous works showed that the solution to the deadlock problem could be provided by proper allocation of buffer capacity and design of the dispatching rules which handle the required synchronisation of the process flows [3, 7, 11]. The methodology presented in previous works enabled to predict the behaviour of a Flexible Manufacturing System, given operation times, buffer capacities, system initial state, and rules that govern the access to the shared resources (i.e. dispatching rules). The results obtained could be treated as an extension of the authors' earlier works, which were limited to simple chain-like topologies of interacting processes [12], to more general structures involving repetitive systems, including complex interactions of closed loops of the cyclic workflow. The modelling approach was based on the concept of *critical resources* (i.e., *system bottlenecks*) and was focused on the development of sufficient conditions for self-synchronised execution of concurrent processes. A number of analytical and simulation models dealing with the problem of prediction and verification of system performance (such as throughput rates and product cycle times) have been developed [1,4]. Many of these models, while effective in modelling various inter-related subsystems, fail when addressing the issue of performance evaluation of a whole system in the case where performance depends not only on the effectiveness of the component elements but also on the synchronisation of their interactions [8]. In other words, attention was concentrated mostly on performance analysis rather than on control. From the control perspective, however, it was more expedient to define a specification for some new desired behaviour, and determine whether the specification could be met through a set of controllable events [5]. Consequently, the problem of distributed system control could be seen as a problem of defining a set of rules (laws) that locally constrain the way the distributed processes flows interact with each other, so as to guarantee a desired performance of a whole system.

In the presented context, this remaining text deals with determining a pair of (an initial state, a set of dispatching rules) that ensures a deadlock-free and starvation-free process flow. Because there are various values of the system resource utilisation rate, which depend on both the pair and the system resource capacity, a procedure of system prototyping is introduced. This result is also a continuation of research described in [2, 9], which focuses at determining sufficient conditions for cyclic behaviour of a system with concurrent processes. Additionally, while investigating the methods of deadlock avoidance presented in [6] conditions that allow creating distributed controlling procedures, which guarantee a cyclic deadlock-free process flow, are also provided.

Finally, we use the procedures mentioned above to present an application where these procedures might be useful in a system performance evaluation.

2. PROBLEM STATEMENT

Before the problem will be formulated, let us consider an illustrative example of connections linking the above mentioned items: a structure of priority selecting rule allocations, choosing an initial state, and an available allocation of system resource capacities. The following notation is used:
$Z(n)$ is a system that consists of n concurrent processes.
$R = \{r_1, r_2, ..., r_q\}$ is the set of resources, R_s, and R_u are subsets of shared and unshared resources respectively, such that $R_s \subseteq R$, $R_u \subset R$, $R_S \cap R_u = \varnothing$, $R_s \cup R_u = R$.

$c = \sum_{i=1}^{q} q_i$, q_i is the capacity of the i-th resource, c is the global resource capacity;

$$\begin{array}{cccccc} & r_1, & r_2, & ..., & r_q & \\ S_j=(& S_j(r_1), & S_j(r_2), & ..., & S_j(r_q) &) \end{array}$$ - the j-th state,

$S_j(r_k)$ – the k-th entry of the j-th state.

$$S_j(r_k) = \begin{cases} p_i \text{ if the i-th process is allocated to the k-th resource,} \\ \Lambda \text{ if no process is allocated to the k-th resource.} \end{cases}$$

p_i is the the i-th process.
$\sigma_i = (p_1, p_2,... ,p_k, ..., p_j, ..., p_m)$ denotes a dispatching rule determining the order the process $p_1, p_2,... ,p_k, ..., p_j, ..., p_m$ can access the i-th resource. So, in this case, after the m-th access to the resource r_i, e.g., after execution of an operation of the process p_m, the next access to r_i is allocated to p_1, then p_2, and so on. A process can occur in a rule once only.
$M_i = (r_1, r_2, ..., r_w)$ - a route of the i-th process (a system resource sequence through that the i-th process goes).
$crd_iA = a_i$ is the i-th entry of the sequence A, $A = (a_1, a_2, ..., a_n)$.
$PR(\sigma_{ri})$ is a set of dispatching rules assigned to the resources directly linked with the i-th resource.
$P(\sigma_{ri})$ is a set of processes occurring in the dispatching rule assigned to the i-th resource.
$F_g(p_h)$ is a number of times the h-th process occurs within dispatching rule assigned to the g-th resource; for instance, $F_4(p_3) = 2$ for a dispatching rule $\sigma_{r4}=(p_2,p_3,p_1,p_3)$.

$\prec$ is the priority relation, $a \prec b \Leftrightarrow h(a) < h(b)$, $h(b) \in N$.

P_i is the i-th major closed loop of resources $P_i = (r_i, r_{i+1}, \ldots, r_j)$, such that each process p_j having in its route a resource $r_t \in R_s$ occurring in the loop P_i, goes through consecutive resources $r_{t+1}, r_{t+2}, \ldots, r_{t+n} \in R_s$ occurring in the loop P_i. The total number of the consecutive resources used by each process must be not less than two resources; for instance, if $P_i = (r_1, r_2, r_3)$, then the route fragments of the processes p_1, p_2, and p_3 may be as follows: $M_1 = (\ldots, r_1, r_2, \ldots)$, $M_2 = (\ldots, r_2, r_3, \ldots)$ and $M_3 = (\ldots, r_1, r_3, \ldots)$.

Whenever in this chapter a closed loop is refereed, it is assumed that the loop is a major one, if not explicitly specified.

PO_i is the i – th minor closed loop of resources $PO_i = (r_i, r_{i+1}, \ldots, r_j)$, that does not meet the conditions for major closed loop P_i.

$RL(P_i)$ is a set of resources occurring in the closed loop P_i,

$RLO(PO_i)$ is a set of resources occurring in the minor closed loop PO_i,

$H_P = (p_i, p_j, \ldots, p_k)$ is a hierarchy vector of processes, such that $\forall P_j$, $\forall PO_k$, $\forall r_j \in RL(P_j)$, $\forall r_j \in RLO(PO_j)$, $\forall \sigma_{ri} = (p_{k1}, p_{r1}, p_{t1}, \ldots, p_{v1})$ $p_{k1}, p_{t1} : p_{k1} \prec p_{t1} \Rightarrow h(p_{k1}) < h(p_{t1})$, $\forall \sigma_{rj} = (p_{k2}, p_{r2}, p_{t2}, \ldots, p_{v2})$ $p_{k2}, p_{t2} : p_{k2} \prec p_{t2} \Rightarrow h(p_{k2}) < h(p_{t2})$, *and* $p_{k1}, p_{k2} : p_{k1} \prec p_{k2} \Rightarrow h(p_{k1}) < h(p_{k2})$.

$H_L = (P_i, P_j, \ldots, P_k, PO_t, PO_u, \ldots, PO_z)$ - a hierarchy vector of closed loops, such that $\forall P_j$, $\forall PO_k$, $P_j \prec PO_k \Rightarrow h(P_j) < h(PO_k)$.

ZP is a set of closed loops.

PZ_i is a set of resources not being allocated in P_i, $PZ_i = \{p_i \mid S_0(r_j) = \Lambda \wedge r_j \in P_i\}$.

$$RRU = \frac{\sum_{i=1}^{n} t_i}{nT}$$

t_i is a time the i-th process spends on a resource within the period T.

n is a number of processes processed within the period T

2.1 Illustrative example

Given a system Z(3). Let $R_s = \{r_2, r_3, r_5\}$ and $R_u = \{r_1, r_4, r_6\}$. Let $\forall i \in \{1, \ldots 6\}$ $q_i = 1$, *so* $c = 6$. The structure of the process routes is shown in Fig. 7-1. Sought is a pair (an initial state, a set of dispatching rules), i.e. (S_0, Θ), which guarantees a cyclic flow of concurrent processes with a given number of copies and a given (expected, required) rate of the system resource capacity.

Let us consider the initial state $S_0 = (p_1, \Lambda, \Lambda, p_2, \Lambda, p_3)$ and the set of dispatching rules $\{\sigma_2 = (p_2, p_1), \sigma_3 = (p_1, p_3), \sigma_5 = (p_3, p_2)\}$. The assumed state leads to a deadlock, in which the process p_1 requests access to the resource r_2, the process p_2 requests access to the resource r_5, and the process

p_3 requests access to the resource r_3. The closed loop of mutual requests is illustrated in Table 7-1.

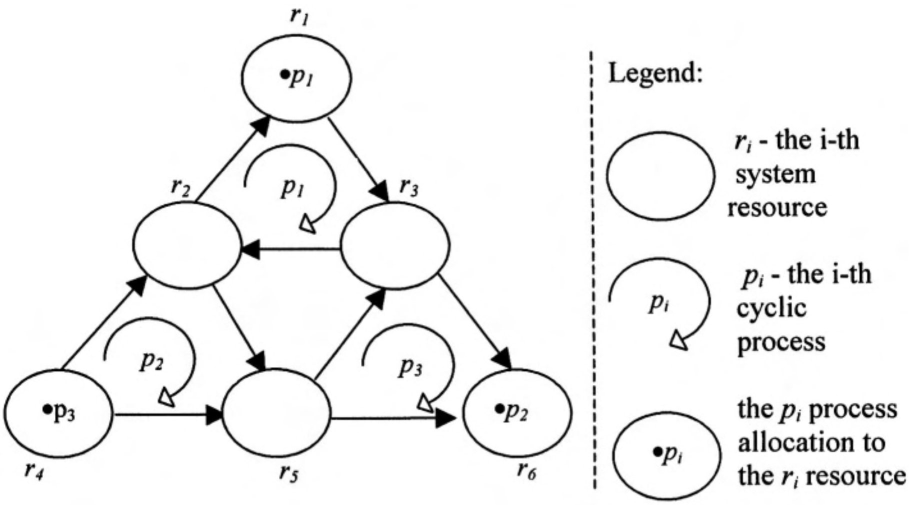

r_1
•p_1
p_1
r_2
r_3
p_2
p_3
•p_3
•p_2
r_4
r_5
r_6
Legend:
r_i - the i-th system resource
p_i
p_i - the i-th cyclic process
•p_i
the p_i process allocation to the r_i resource

Figure 7-1. An example of a system consisting of three processes.

p_i* means that the i-th process occupying the relevant resource is awaiting the next resource (i.e. determined by the i-th process[1] route), p_i means that the i-th process is executed on the resource, Λ means that no process is executed nor awaiting on the resource.

It is possible to find either another pair (S_0, Θ) for which the system is deadlock-free, or to increase the system resource capacity. In both these a rate of the system resource utilisation may change:

Table 7-1. The sequence of states reached from the state S_0 for the pair ($S_0 = (p_1, \Lambda, \Lambda, p_2, \Lambda, p_3)$, $\Theta = \{\sigma_2 = (p_2, p_1), \sigma_3 = (p_1, p_3), \sigma_5 = (p_3, p_2)\}$).

	S_0	S_1	S_2
r_1	p_1	Λ	Λ
r_2	Λ	p_2	p_2*
r_3	Λ	p_1	p_1*
r_4	p_2	Λ	Λ
r_5	Λ	p_3	p_3*
r_6	p_3	Λ	Λ

a) Let us change the set of dispatching rules so that they were equal to, $\Theta = \{\sigma_2 = (p_1, p_2), \sigma_3 = (p_1, p_3), \sigma_5 = (p_2, p_3)\}$. Note that the initial state and capacity of the system resources remains unchanged. Table 7-2, containing the sequence of states for that set of dispatching rules, shows that the process flow is cyclic and deadlock-free.

Table.7- 2. The sequence of states reached from the state S_0 for the pair ($S_0 = (p_1, \Lambda, \Lambda, p_2, \Lambda, p_3)$, $\Theta = \{\sigma_2 = (p_1, p_2), \sigma_3 = (p_1, p_3), \sigma_5 = (p_2, p_3)\}$).

	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_2	S_3	S_4	S_5	S_6	S_7
r_1	p_1	Λ	Λ	p_1	p_1*	p_1*	p_1*	Λ	Λ	p_1	p_1*	p_1*	p_1*	Λ
r_2	Λ	Λ	p_1	p_2	Λ	Λ	Λ	Λ	p_1	p_2	Λ	Λ	Λ	Λ
r_3	Λ	p_1	Λ	Λ	Λ	Λ	p_3	p_1	Λ	Λ	Λ	Λ	p_3	p_1
r_4	p_2	p_2*	p_2*	Λ	Λ	p_2	p_2*	p_2*	p_2*	Λ	Λ	p_2	p_2*	p_2*
r_5	Λ	Λ	Λ	Λ	p_2	p_3	Λ	Λ	Λ	Λ	p_2	p_3	Λ	Λ
r_6	p_3	p_3*	p_3*	p_3*	p_3*	Λ	Λ	p_3	p_3*	p_3*	p_3*	Λ	Λ	p_3

The presented sequence of process allocations shows that there are a transit period (denoted by two-state sequence S_0-S_1), and then a steady state period denoted by the sequence: S_2, S_2, ..., S_7. From the table it follows also that ***RRU*= 42,8 *%*** (at $c = 6$).

b) Let us add an additional capacity to the resource r_2. Since the additional capacity may be interpreted as room for an additional process, the resource r_2 is capable of allocating two processes simultaneously. Because $q_2 = 2$, there is no need to assign a priority selecting rule to that resource r_2. As a result the set of dispatching rules may be as follows: $\Theta = \{\sigma_2 = (),$ $\sigma_3 = (p_1, p_3),\ \sigma_5 = (p_2, p_3)\}$. Figure 7-2 contains the system with the capacity of the resource r_2 equals to 2. Table 7-3, containing the sequence of states, shows that the processes flow is cyclic and deadlock-free. The presented sequence of process allocations shows that there is a transitional period (denoted by three-state sequence from S_0, to S_4), and a period denoted by the sequence: S_5, S_6, S_7. From the table above it follows also that ***RRU*= 100 *%*** (at $c = 7$)This means that thanks to increasing the number of resources in the overall system by one unit, the rate of the system resource utilisation has increased from 42,8% to 100%.

c) Let us use in this case the system shown in Figure 7-1, that is $\forall i \in \{1,\ldots 6\}\ q_i = 1$, so $c = 6$. Table 7-3, containing the sequence of states, shows that the processes flow is cyclic and deadlock-free for the following pair (an initial state, a set of dispatching rules): ($S_0 = (\Lambda, \Lambda, p_1, \Lambda, p_2, p_3)$, $\Theta = \{\sigma_2 = (p_1, p_2),\ \sigma_3 = (p_1, p_3),\ \sigma_5 = (p_2, p_3)\}$). From Table 7-4 it follows also that ***RRU*= 100 *%*** (at $c = 6$). Thus, although the total system capacity is less than in the case b) (it is the same as in the case a, where *RRU* was 42,8%), *RRU* still reaches the value of 100%. The reason is another initial state chosen for the case c).

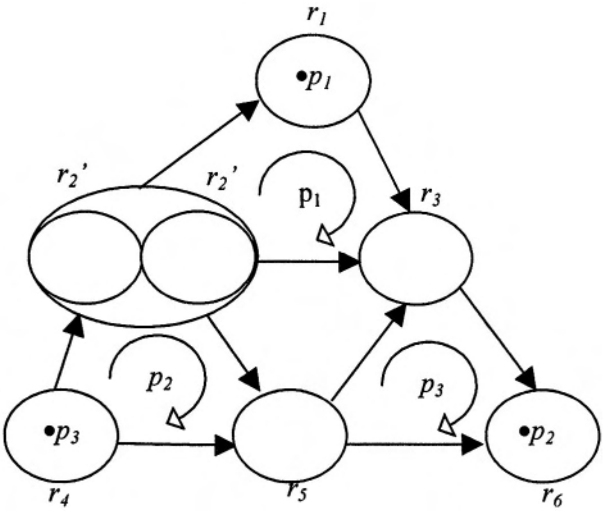


Figure 7-2. An example of a system consisting of three processes with the capacity of the resource $\mathbf{r_2}$ equals to 2.

Table 7-3. The sequence of states reached from the state S_0 for the pair ($S_0 = (p_1, \Lambda, \Lambda, p_2, \Lambda, p_3)$, $\Theta = \{\sigma_2 = (), \sigma_3 = (p_1, p_3), \sigma_5 = (p_2, p_3)\}$).

	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_5
r_1	p_1	Λ	Λ	p_1	p_1*	Λ	Λ	p_1	Λ
r_2	Λ	p_2	p_1	Λ	p_2	Λ	p_1	p_2	Λ
r_3	Λ	p_1	Λ	Λ	p_3	p_1	Λ	p_3	P_1
r_4	p_2	Λ	Λ	p_2	Λ	Λ	p_2	Λ	Λ
r_5	Λ	Λ	p_2	p_3	Λ	p_2	p_3	Λ	P_2
r_6	p_3	p_3*	p_3*	Λ	Λ	p_3	Λ	Λ	P_3

From the observations above it follows that the parameters, which determine the process flow are a system period *T* and a rate of the system resource utilisation. Therefore, a system flow is restricted by its period *T,* a pair (S_0, Θ) and the system resource capacity *c*. Thus, as it was also shown in the mentioned cases, the system flow, and the value of *RRU* as well, may be controlled by three factors: the system capacity, an initial state and a set of dispatching rules. Finally, it is worth to stress that the pair guarantees that the system has a cyclic, deadlock-free flow.

Table 7-4. **The sequence of states reached from the state S_0 for the pair ($S_0 = (\Lambda, \Lambda, p_1, \Lambda, p_2, p_3)$, $\Theta = \{\sigma_2 = (p_1, p_2), \sigma_3 = (p_1, p_3), \sigma_5 = (p_2, p_3)\}$).**

	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_5
R_1	Λ	Λ	p_1	Λ	Λ	p_1	Λ	Λ	p_1
R_2	Λ	p_1	p_2	Λ	p_1	p_2	Λ	p_1	p_2
R_3	p_1	Λ	p_3	p_1	Λ	p_3	p_1	Λ	p_3
R_4	Λ	p_2	Λ	Λ	p_2	Λ	Λ	p_2	Λ
R_5	p_2	p_3	Λ	p_2	p_3	Λ	p_2	p_3	Λ
R_6	p_3	Λ	Λ	p_3	Λ	Λ	p_3	Λ	Λ

2.2 Problem definition

The problem discussed in this chapter may be divided into two issues. The first one links a possibility of determining a pair (S_0, Θ) for a class of system consisting of major and minor loops (see Section 2.1). Such a pair ensures that there is no deadlock in the system. Furthermore, the capacity of each resource may be equal to 1 (what can be interpreted as cost reduction). The second issue of the problem deals with a rate of the system resource utilisation. Both these parts are described more detailed below.

Issue 1. In the considered system class, with an assumption of a given structure of capacity allocation, a rule to synchronise the access to the resources by the processes, requires, in general, to choose an initial, acceptable allocation of the processes to the resources. The initial allocation is required, because some arbitrary process allocations, along with the rules limiting access to the resources, may lead to a deadlock.

In turn the number of processes that may be executed simultaneously is conditioned by three factors: a value of an acceptable initial state, the system resource capacity allocation, and dispatching rules. This means that using permanent (unchanged in time) process synchronizing rules guarantees the required process flow as well as assures that the flow is repetitive (synchronized).

In general case, however, both dispatching rules and an initial processes allocation have to be selected properly as to avoid a deadlock. It means each situation corresponding to activation of a new process calls for the proper examination of an initial state (i.e. initial processes allocation) and a set of dispatching rules assigned (see Table 7-1).

Let us consider a main problem formulated as follows: Given is a system of concurrent cyclic processes, i.e. the routes of the processes and resource capacity allocations are known. Sought are such conditions, which would be held by a pair (S_0, Θ) to guarantee a cyclic process flow.

Theorem

Consider a system of cyclic processes, and a given pair $(S_{0,} \Theta)$, where $S_0 = (s_1, s_2, \ldots, s_q)$, $\Theta = \{\sigma_{rj} = (p_k, p_h, \ldots, p_i, \ldots, p_v) \mid r_j \in R_s\}$. Assume $H_L = (P_i, P_j, \ldots, P_k, PO_t, PO_u, \ldots, PO_z)$, $H_P = (p_i, p_j, \ldots, p_k)$. If the following conditions hold, then the system has a cyclic steady state for (S_o, Θ).

i) $\forall \sigma_{ri} \in PR(\sigma_{ri}), p_d \in P(\sigma_{ri}) : F_i(p_d) = F_k(p_d)$

ii) $\forall r_i \in R_s, \ \forall \sigma_{ri} = (p_k, p_r, p_t, \ldots, p_v) \ \ p_k, p_t : p_k \prec p_t \Rightarrow h(p_k) < h(p_t)$

iii) $\forall P_i \in ZP, \exists PZ_i = \{p_i \mid S_0(r_j) = \Lambda \wedge r_j \in P_i\} : |PZ_i| \geq 1$

iv) $\forall P_i, \ \forall r_i \in RL(P_i), (S_0(r_i) = p_k) \Rightarrow (p_k = crd_1 \sigma_{ri}) \ \& \ \forall r_i \notin R_s, (S_0(r_i) = p_k) \Rightarrow (p_k \notin \cup\{crd_1 \sigma_{ri} \mid r_i \in R_s\})$

v) $\forall PO_i, \ \forall r_i \in RLO(PO_i), (S_0(r_i) = p_k) \Rightarrow (p_k = crd_1 \sigma_{ri}) \ \& \ crd_1 M_k = r_i$

vi) $\forall p_k \in \cup\{P(\sigma_{ri}) \mid r_i \in R_s\}, \exists! r_i \in R, S_0(r_i) = p_k$

vii) $|\{S_0(r_i) \neq \Lambda \mid r_i \in R\}| = |\cup\{P(\sigma_{ri}) \mid r_i \in R_s\}|$

The condition i) guarantees that each process occurs the same number of times in a priority selecting rule assigned to each shared resource, which the process has in its route. The condition ii) ensures that the order of processes occurring in the dispatching rules assigned to the resources is ordered according to the priority relation, i.e. the positions of processes in the process hierarchy vector. The condition iii) ensures that in each major closed loop at most n-1 shared resources is allocated in an initial state. The next condition iv) guarantees that a process going through a major closed loop is allocated in an initial state on the shared resource occurs on the first co-ordinate in the priority selecting rule assigned to the shared resource, whereas a process that is allocated in the initial state on a non-shared resource, never occurs at the first co-ordinate of any priority selecting rule assigned to shared resources. The condition v) ensures that on a resource belonging to a minor closed loop may be allocated in an initial state only a process that occurs on the first position of the rule assigned to that resource and which has that resource as the first resource in its route. The condition vi) guarantees that each process is allocated exactly once in the initial state and finally, the condition vii) ensures that the number of process allocations in the initial state is equals to the number of processes in the system.

For the proof let us observe that the condition i) guarantees that there is no recurrence in the processes' routes. The next condition, ii), ensures that the process p_k must not be allocated prior to the process p_i to a shared resource, if $p_i \prec p_k$ because the processes are ordered according to their occurrence in major and minor closed loops (where the major loops are of higher hierarchy - see the definition of the loop hierarchy). From the above it follows that if the processes p_i and p_k have two or more shared resources in their routes, the order of these processes in the dispatching rules assigned to the shared resources guarantees lack of mutual process blockade. The condition iii) makes that there is no process blockade within the loop, while thanks to the condition iv) the process allocated on the non-shared resource in the initial state does not cause a possible deadlock by allocation itself on the shared resource(s) belonging to a closed cyclic loop. Such a deadlock would be possible, if any processes not allocated in the initial state on any of the resources of a closed cyclic loop were able to be allocated on a resource belonging to the major closed loops prior other processes allocated in the initial state on resources that forms a major closed loop. This condition implies also that any process not allocated in the initial state on a resource of a major closed loop, has a less priority than the processes allocated in the initial state on resources belonging to a major closed loop. Because on a resource belonging to a minor closed loop may be allocated in an initial state only a process that occurs on the first position of the rule assigned to that resource and which has that resource as the first resource in its route the condition v) guarantees that other processes do not block that process. Finally, the conditions vi) and vii) ensures that no process is allocated on resources two or more times in the initial state and that no process is skipped in that state.

Taking into account that the processes are prioritised and that at least one resource of each major closed loop remains unallocated in the initial state, the processes allocated in the initial state on resources belonging to a major closed loop may allocate the next resources in their routes without any interruption of other processes. Thus, the processes may go through major closed loops not being suspended by other processes. Once a process leaves a loop, it enters a "safe" area (i.e. a resource). Then the other processes may be allocated on resources according to their routes. As a result there is no process blockade, so the system has cyclic steady state.

For example, in the case of the system from Figure 7-1, the following pair (S_0, Θ), and the hierarchy $H_L = (P_1 = (r_2, r_3, r_5), PO_1 = (r_1, r_3, r_6, r_5, r_4, r_2))$, $H_p = (p_1, p_3, p_1)$, where $S_0 = (\Lambda, \Lambda, p_1, p_2, p_3, \Lambda)$, $\Theta = \{\sigma_2 = (p_1, p_2), \sigma_3 = (p_1, p_3), \sigma_5 = (p_3, p_2)\}$ lead the system to a cyclic steady state – see Table 7-5.

Form Table 7-5 it follows that the system shown in Figure 7-1 has a cyclic steady state for the pair $(S_0 = (\Lambda, \Lambda, p_1, p_2, p_3, \Lambda), \Theta = \{\sigma_2 = (p_1, p_2), \sigma_3 = (p_1, p_3), \sigma_5 = (p_3, p_2)\})$, which meets the conditions provided by Theorem.

Table 7-5. The sequence of states reached from the state S_0 for the pair $(S_0 = (\Lambda, \Lambda, p_1, p_2, p_3, \Lambda), \Theta = \{\sigma_2 = (p_1, p_2), \sigma_3 = (p_1, p_3), \sigma_5 = (p_3,p_2)\})$.

	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_6	S_7	S_8
r_1	Λ	Λ	p_1	Λ	Λ	p_1	Λ	Λ	P_1	Λ
r_2	Λ	p_1	p_2	Λ	p_1	p_2	Λ	p_1	P_2	Λ
r_3	p_1	p_3	Λ	p_1	Λ	p_3	p_1	Λ	P_3	p_1
r_4	p_2	p_2^*	Λ	Λ	p_2	Λ	Λ	p_2	Λ	Λ
r_5	p_3	Λ	Λ	p_2	p_3	Λ	p_2	p_3	Λ	p_2
r_6	Λ	Λ	p_3	p_3^*	Λ	Λ	p_3	Λ	Λ	p_3

Issue 2. The other issue links the system resource capacity and a rate of the system resource utilisation. Note that usually the less is the system resource capacity the less is a rate of the resource utilisation in the system. Thus, when the system becomes cheaper (because of the resource reduction) the rate of the resource utilisation goes down as well. This implies that there is a need to choose between the system resource capacity *(q)* and the rate of the system resource utilisation. One of these factors may be sought when the other remains constant, i.e.:

What a pair (S_0, Θ) guarantees that q reaches its minimal value, assuming that *RRU* is constant?

What a pair (S_0, Θ) guarantees that *RRU* reaches its maximum value, assuming that q is constant?

Although the conditions given in Theorem guarantee that there is a cyclic steady state flow in the systems consisting of resources with their capacities equals to one unit each, there is a possibility to increase the resource capacities to reach a higher value of *RRU* in some cases.

3. APPLICATION

Let us consider a system composed of a set of concurrently executed cyclic processes. Each process flow is specified by a sequence of system resources. Resources are specified by they capacities, i.e. ability to process simultaneously one or more processes.

Conditions obtained allow one to consider the following two algorithms:

Consider a system of cyclic processes. Assume $H_L = (P_i, P_j, \ldots, P_k, PO_t, PO_u, \ldots, PO_z)$, $H_P = (p_i, p_j, \ldots, p_k)$, and $S_0 = (s_1, s_2, \ldots, s_q)$. So, if the conditions of Theorem hold, then there exists $\Theta = \{\sigma_{rj} = (\ldots, p_h, \ldots, p_j, \ldots, p_q, \ldots) \mid r_j \in R_s\}$ such that the system has a cyclic steady state.

Consider a system of cyclic processes. Assume $H_L = (P_i, P_j, \ldots, P_k, PO_t, PO_u, \ldots, PO_z)$, $H_P = (p_i, p_j, \ldots, p_k)$, and $\Theta = \{\sigma_{rj} = (\ldots, p_h, \ldots, p_j, \ldots, p_q, \ldots) \mid r_j \in R_s\}$. If the conditions of Theorem hold, then there exists $S_0 = (s_1, s_2, \ldots, s_q)$ such that the system has a cyclic steady state.

First of them enables one searching for the priority dispatching rules assignment, while the second one for a safe an initial processes allocation.

For illustration of the first case see Table 7-5, which shows the process flow of the system from Figure 7-1 for the assumed initial state $S_0 = (\Lambda, \Lambda, p_1, p_2, p_3, \Lambda)$ and a found set of dispatching rules $\Theta = \{\sigma_2 = (p_1, p_2), \sigma_3 = (p_1, p_3), \sigma_5 = (p_3, p_2)\}$. The transient period consists of four states, i.e. S_0-S_3, whereas the steady state period contains three states - from S_4 to S_6.

Both of the algorithms, however, are aimed at an alternative production flow control prototyping. In order to illustrate the possible applications let us consider the system shown in Figure 7-3. The task regards of system performance evaluation for different variants of resources capacity allocation. As a performance index, a rate of the system resource utilisation is used.

Assume $S_0 = (A, p_3, A, p_1, A, A, p_2)$. Consider the following three variants of system capacity allocation:

1. $q_1 = q_2 = q_3 = q_4 = q_7 = 1, q_5 = q_6 = 2$, i.e. $c = 9$;
2. $q_1 = q_2 = q_3 = q_4 = q_5 = q_7 = 1, q_6 = 2$, i.e. $c = 8$;
3. $q_1 = q_2 = q_3 = q_4 = q_5 = q_6 = q_7 = 1$, i.e. $c = 7$.

In the first case the system has a cyclic steady state for any assignment of dispatching priority rules $\Theta = \{\}$, and $RRU = 81\%$. In the second case the system has a cyclic steady state for $\Theta = \{\sigma_1=\sigma_4=(p_1, p_3)\}$, and $RRU= 67\%$. Finally, for the last case the system has a cyclic steady state for $\Theta = \{\sigma_1=\sigma_4=(p_1, p_3), \sigma_3=\sigma_6= (p_2, p_3)\}$, and $RRU = 61\%$. The results are shown in Figure 7-4.

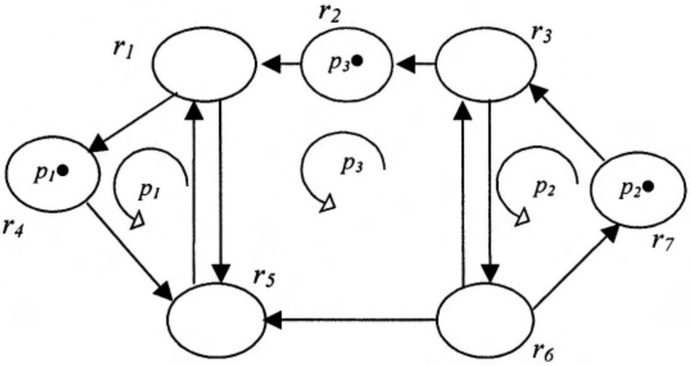


Figure 7-3. A system of concurrent cyclic processes.

So, the production flow control prototyping can be seen as a one of the following problems:

a)

$RRU \rightarrow MAX$

s.t.

$c = \gamma(RRU, (S_0, \Theta)) \leq const.$

For assumed c find out such (S_o, Θ) that $RRU \rightarrow MAX$

b)

$c \rightarrow MIN$

s.t

$RRU = \varphi(c, (S_0, \Theta)) \geq const.$

For assumed RRU find out such (S_o, Θ) that $c \rightarrow MIN$

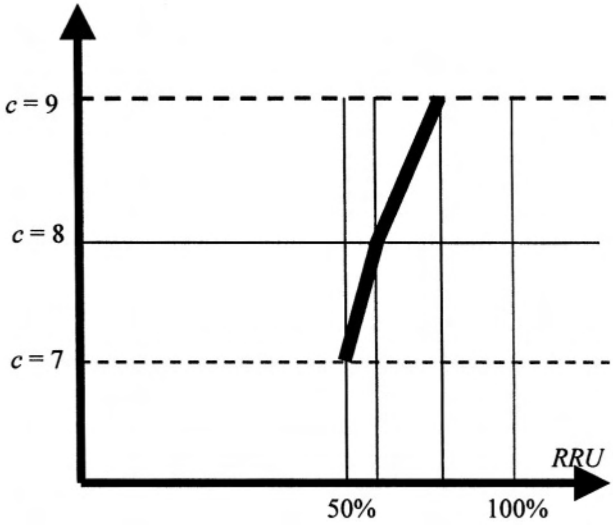


Figure 7-4. System prototyping from RRU perspective.

It is easy to note that solutions may be found through heuristic searching. Therefore the software tools supporting the decision maker are of pivotal importance. They development in turn depends on both: automatic examination whether the control assumed is admissible and if so on an automatic evaluation (free of computer simulation) of system performance measures. It means, that the results already obtained should be extended for the conditions guaranteeing that an initial state S_0 from the pair (S_0, Θ) belongs to the states occurring in a cyclic steady state.

The max-plus algebra formalism can be applied in order to determine of system performance indices, as the shared resources usage rate or the process realization rate, in an algebraic manner. Such an operation would be performed automatically, basing on the desired system performance (or the values of given indices) and the system structure. As a result it is feasible to determine a set of alternative production flows meeting the assumed values of the rates and system capacity.

4. CONCLUSIONS

The presented conditions allow creating the distributed controlling procedures in systems of concurrent processes. This means that considering a system, which guarantees that for an arbitrary chosen pair (S_0, Θ) the system is deadlock-free [10], it is possible to determine alternative process flows, which guarantee an excepted system performance. Therefore they provide a framework for development of automated production flows and finally work flows prototyping. Such a process of system prototyping may take advantage of the max-plus algebra formalism, which – by applying the procedure of an analytical model synthesis – allows determining the values of various system performance indices, such as the system resource usage, process realizations and the system period for a given structure. Future research will deal with this task.

REFERENCES

1. Baccelli F.G., et al., 1992. Synchronization and linearity. In: *An algebra for discrete event systems,* John Wiley & Sons, Chichester.
2. Banaszak Z., Polak M., 2002, Buffers vs. Initial State and Priority Rules Allocation, *Proceedings of the 8th International Symposium on „ Methods and Models in Automation and Robotics ",* Kaszyński R. Ed., 2-5 September, Szczecin, Poland, Vol. 2, 1109-1114
3. Banaszak Z.A., Polak M., 2002, Deadlock-free distributed control for repetitive flows, *Proceedings of the 6th Int. Workshop on Discrete Event Systems,* October 2-4, Zaragoza, Spain, 273-278.
4. Camus, H., H. Ohl, O. Korbaa and J.C. Gentina 1996, Cyclic schedules in flexible manufacturing systems with flexibilities in operating sequences. *Proceedings of the Workshop on Manufacturing and Petri Nets,* Osaka, Japan, 97-116.
5. Fanti, M.P., B. Maione, G. Piscitelli and B. Turchiano, 1996. System approach to design generic software for real-time control of flexible manufacturing systems. *IEEE Trans. on Systems, Man, and Cybernetics.* 26, 190-202.
6. Lawley M.A., Reveliotis S.A., Ferreira P.M., 1998, A correct and scalable deadlock avoidance policy for flexible manufacturing systems. *IEEE Trans. on Robotics and Automation,* 14, 796-809.
7. Lee, T. and J. Song 1996. Petri net modeling and scheduling of periodic job shops with blocking. *Proceedings of the Workshop on Manufacturing and Petri Nets,* Osaka, Japan, 197-214.
8. Ramamritham, K. 1995. Allocation and scheduling of precedence-related periodic tasks. *IEEE Trans. on Parallel and Distributed Systems,* 6, 412-420.
9. Spiridon A. Reveliotis, Mark A. Laweley, Placid M. Ferreira, 1997, Polynomial-complexity deadlock avoidance policies for sequential resource allocation systems. *IEEE Trans. on Automatic Control,* 42,, 1344-1357
10. Zaremba M.B., Banaszak Z.A., Majdzik P., Jędrzejek K.J., 1999, Distributed Flow control design for repetitive manufacturing processes. *Proceedings of the 14th Triennial Worlds Congress ofIFAC,* July 5-9, Beijing, P.R. China, A, 1-6.
11. Zaremba M.B., K.J. Jedrzejek and Z.A. Banaszak 1998. Design of steady-state behaviour of concurrent repetitive processes: An algebraic approach. *IEEE Trans. on Systems, Man, and Cybernetics,* 28, 199-212
12. Zaremba M.B. and Z.A. Banaszak, 1995. Performance evaluation for concurrent processing in cyclic systems. *Concurrent Engineering: Research andApplications,* 3, 123-130.

Chapter 8

MODELLING OF THE SUPPLY CHAIN FOR A DISTRIBUTED PUBLISHING ENTERPRISE

Oleg Zaikin, Przemysław Korytkowski,
Emma Kushtina, Bartłomiej Małachowski

Abstract: This chapter is devoted to one of the main problems in serving external orders in a corporate network of publishing services. This problem was already examined in our previous publications, but only in the case of a star network structure. In this chapter, the problem of total production cost minimisation for a general network structure, given by a digraph, is examined.

Key words: publishing supply chain, modelling and simulation, performance optimisation.

1. INTRODUCTION

In constantly developing publishing industry the modern and high efficient prepress and press centres provide better quality of a final product and reduction of all material costs (paper, film etc.). Good examples of such equipment are: Heidelberg Speedmaster 74, Quickmaster DI 46-4 Pro [6].

However, a price of this equipment amounts to over one million EUR, which bound its accessibility on the wide market of the printing and publishing services. As it was shown in [3, 9], such situation leads to necessity to organise distributed publishing enterprise in which expensive and flexible printing equipment are allocated within a *network of publishing services*. Such a network has an ability to fulfil a lot of diverse clients' needs regarding type, print run and quality of the publishing product.

Design and organization of such a network needs a new approach, because traditional specification of a publishing product using the features as format, colour type, raster density proved to be insufficient. These features are decisive in a production organization process, which is oriented on serving orders for one type of product with a specified cost and quality level. In traditional approach, basing on these features of the printing product, a set of necessary devices, publishing cost and time can be clearly defined. In the case of introducing a new technology based on flexible printing pieces of equipment (flexible printing devices), the features mentioned above are equally important. However, cost and time of servicing depend also on another group of features like: edition personalisation ability, intensity of products ordering, volume of an edition of each type, ratio of order types for all printing activities, etc. Thus, transition from separated printing company which is based on highly specialised printing devices to corporate network of printing services is similar to transition from product-oriented to process-oriented production.

1.1 Stochastic stream of publishing products and services

Time, number and set of orders for publishing products and services have a random character. In order to produce a product or serve a customer order some standard operations are used. Therefore a stream of orders for every kind of production or services is stochastic, multi-product and its servicing has many phases.

In publishing, different types of products are produced (books, magazines, calendars, etc.), each product needs a preliminary specification of works. Moreover, a specification of one type of product belonging to different orders could vary significantly, considering a necessary set of works, their volume and quality requirements.

For example for two different books, a volume of work needed to a text processing can differ many times. Further, quality requirements of illustrations for a book can also differ; one book can have only black and white drawings, another one only colour photographs.

From the point of view of the specification, there is a big difference between a product and a service. A service is an order for preparing not a whole product, but only one specific job and therefore it does not need preliminary specification (for example ‘file colour separation’). Usually this type of orders is random and at the same time has very precise quality requirements.

Thus, stochastic character of the order stream results from the following factors:

– time of order arrival,
– its volume,
– unpredictable result of ordered product specification.

1.2 Transforming a stream of orders into a stream of elementary works

For each planning period, there are a lot of orders, which require a set of works for their products. The final set of works to perform in the current period is a result of the period orders and their preliminary specification (decomposition into works). The sequence of their execution is determined by technological route. Different works can require the same device at the same time. The process of transforming orders for products into orders for works was examined in [10]. The general schema of relations between different data, which influence specifications of every order, is shown in Figure 8-1.

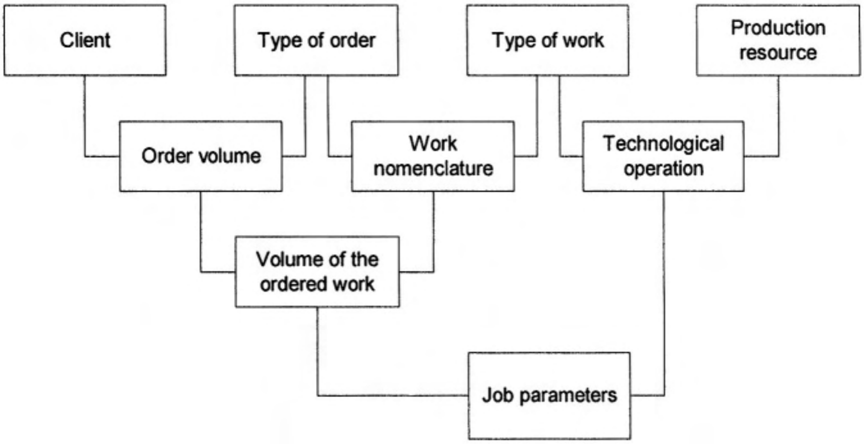


Figure 8-1. General schema of an order specification

In the Figure 8-1 the relation 'client – order volume – type of order' shows information about an input stream of orders. The relation 'order type – work nomenclature – type of work' gives information about typical jobs, which need to be done for every order. The relation 'type of work – technological operation – production resource' describes an assignment of typical works to a specific production resource. The relation 'Order volume – volume of the ordered work – work nomenclature' gives a volume of only these works, which are included in a specification of an ordered edition. Finally, relation 'volume of the ordered work – job parameters – technological operation' gives parameters of all necessary jobs, which have to be done in order to serve a received order.

This schema presents all static rules to transform a stream of orders into a stream of elementary works. Dynamic characteristics (rate of arrival, probability distribution) will be considered in the mathematical model of workflow in next section. Using static rules, dynamic characteristics and technological routes allows to design a mathematical model to Supply Chain (SC) performance evaluation.

2. PERFORMANCE OPTIMISATION IN PUBLISHING SUPPLY CHAIN

2.1 Problem statement

A result of the process described above is a superposition of several stochastic input streams oriented on using a set of production resources in a corporate network of publishing services. In this condition two characteristic situations can appear:

- a lack of production capacity to serve all works in due time,
- production resources are not utilised enough (there is too much production capacity).

Because advanced flexible printing devices are very expensive their inadequate utilisation is a very significant loss for a corporation. On the other hand, a lack of production capacity can cause serious delays in a production process (lead-time increase), which also leads to a financial loss (Work in Progress cost, contract penalties, etc.). Thus, the main problem of design and management of a corporate network of publishing services is to find an optimal utilisation for every production resource, which provides minimal total cost equal to the sum of *work in progress* and *resource utilisation costs.*

Therefore the main features of the publishing technological process:

a)decomposition of the customers' orders into a set of works depending on many parameters,

b)stochastic character of the works entering the technological process,

c)stochastic character of time and quality of the technological operations performance,

d)execution of several works belonging to various orders into the same network node,

e) possibility to quickly transfer a large volume of works from one node to another (using computer networks).

A possible method of study of such process is the queuing modelling and simulation [7, 8]. For this approach, the publishing SC is considered as a queuing model with a set of servicing nodes, servicing one or several workflows of customers' demands. Each servicing node performs one kind of technological operation. Each workflow is described by a sequence of technological operations to be processed.

2.2 Performance evaluation model

The input data needed for performance evaluation of SC, are the following:

1. Customers' orders for publishing product and their specification

L is the number of existing customer order types,

$l = 1,...L$ is a customer order type,

K is the number of all possible printing works,

$k = 1,...K$ is a printing work,

$$q_{kl} = \begin{cases} 1, & \text{if order type } 'l' \text{ needs the work type } 'k' \\ 0, & \text{otherwise.} \end{cases}$$

$\overline{\mathrm{H}} = (\eta_l)$ is the vector of customer orders' intensity (rates),

$A = [a_{lk}]$ is the specification matrix of order's, where a_{lk} is average volume of the work of type k for an order of type l.

Using the intensity vector $\overline{\mathrm{H}}$ and specification matrix A we can compute the following characteristics of input streams of works:

$\lambda_k = \sum_{l=1}^{L} \eta_l q_{kl}$ - average rate of arrivals of works,

$v_k = \sum_{l=1}^{L} \eta_l a_{lk}$ - average batch capacity at work of type k .

2. Workflow processes and their parameters

Under the term of a workflow process we understand a stream of works. Let's suppose that each work in the workflow process represents a batch of job units (set of some type of pages, related to the order). Besides let every workflow process be stationary and characterised by a kind of arrival pattern and a kind of the batch capacity. We use the following notation:

$F = \{f_k\}$ is the set of workflow processes,

λ_k is the average rate of arrivals for the workflow 'f_k',

X_k is the kind of arrival pattern for the workflow 'f_k',

v_k is the average batch size (the number of job units for a work of 'f_k'),

Y_k is the distribution of the batch size for the workflow 'f_k'.

3. SC structure

SC structure may be represented as a digraph $G = \{N,L\}$. Vertexes of the digraph $N = \{n_i\}$ are the servicing nodes and arcs of the digraph $L = \{l_j\}$ are the links (channels) connecting them.

4. Parameters of a servicing node

Each servicing node n_i consists of a set of one-type servers s_i, performing some kind of technological operations.

We use the following notation:

μ_i is the rate of servicing (the number of job units processed by the server per time unit),

τ_i is the time of servicing of a job unit,

$\mu_i = 1/\tau_i$.

We consider that input buffers at the servicing nodes of publishing SC are unlimited (works input data are in electronic form).

5. Technological route

Each workflow process $k = 1,...,K$ is being carried through a route of servicing nodes, representing a chain of technological operations $W_k = (n_{k1}, n_{k2}, ..., n_{kj}, ...)$, where n_{kj} are the index of servicing node in SC, j=1,..., J_k is the index of technological operation.

We suppose that all technological routes are sequential and have no loops. In the general case, one servicing node can enter several technological routes corresponding to different workflow processes. Therefore, for modelling the relation between the set of servicing nodes $n_i \in N$ and the set of workflow processes $f_k \in F$ of a SC we use:

$$r_{k\,i} = \begin{Bmatrix} 1, if \quad n_i \in W_k \\ 0, otherwise \end{Bmatrix},$$

where $k = 1,2,...,K, i = 1,2,...,I$, I is the total number of nodes.

Decision variables

Each servicing node can be represented as a multi-server queuing system. Such structure permits realising the parallel servicing process for several works simultaneously. Therefore, we have to define the number of servers, operating in parallel for each servicing node, i.e. $\overline{P} = (P_1, P_2, ..., P_I)$ is the vector of decision variables, where P_i is the number of uniform parallel servers for the servicing node n_i.

Criterion function

We consider two components:

1. Work in progress cost, i.e. total cost caused by flow time (delay),

$$C_1 = \alpha \sum_{n_i \in N} T_i = \alpha \sum_{n_i \in N} \sum_{f_K \in F} r_{k_i}(\tilde{\tau}_i^w + \tau_i^s)\lambda_k v_k,$$

where

T_i is a summary 'flow-time' for streams arriving at the servicing node n_i per time unit,

α is the cost of Work in Progress (WIP) per time unit,

$\tilde{\tau}_i^w$ and τ_i^s are the average waiting and servicing time correspondingly of a job unit (a page) at the servicing node n_i .

2. *Resource utilisation cost,* i.e. the total cost of utilisation of the all servers for a time unit,

$C_2 = \sum_{n_i \in N} P_i(\beta_i \rho_i + \gamma_i(1-\rho_i))$, where

β_i is the unit time processing cost and γ_i is the idle-time cost for a server per time unit of the servicing node n_i,

$\rho_i = \sum_{f_k \in F} r_{ki} \frac{\lambda_k}{P_i \mu_i}$ is the rate of server utilisation for n_i .

If the number of parallel servers P_i increase, then the cost C_1^i (cost component C_1 for the node n_i) linked with work in progress decreases and the cost C_2^i (cost component C_2 for the node n_i) caused by equipment cost increases. Therefore, it is possible to define such values of decision variables $\overline{P} = (P_1, P_2, ..., P_I)$ that provide a minimum value of the total cost:

$$CR = C_1 + C_2 = \min . \qquad (1)$$

Note that the first component of criterion function C_1 defines also the *delay* which must be minimised from the customer's point of view; the second one C_2 defines the *investments* which must be minimised from the network manager's point of view. These components of criterion function (1) depend on the decision variables $\overline{P} = (P_1, P_2, ..., P_I)$ in the opposite way.

The formulated problem refers to the discrete programming one with a non-linear criterion function, depending on a set of stochastic and deterministic variables.

3. OPTIMISATION ALGORITHM

The formulated problem is a discrete optimisation one, which can be solved on the basis of an enumeration algorithm. An algorithm of this type for networks of star configuration is proposed in [3,9]. However, for a large-scale network of general structure this method cannot be used. There is an heuristic algorithm, proposed in [10]. It is based on searching for the most loaded servicing node called the 'bottleneck'. In the model [10] the bottleneck is the servicing node which has the maximal value of the criterion function (1).

The optimisation algorithm is shown in Figure 8-2.

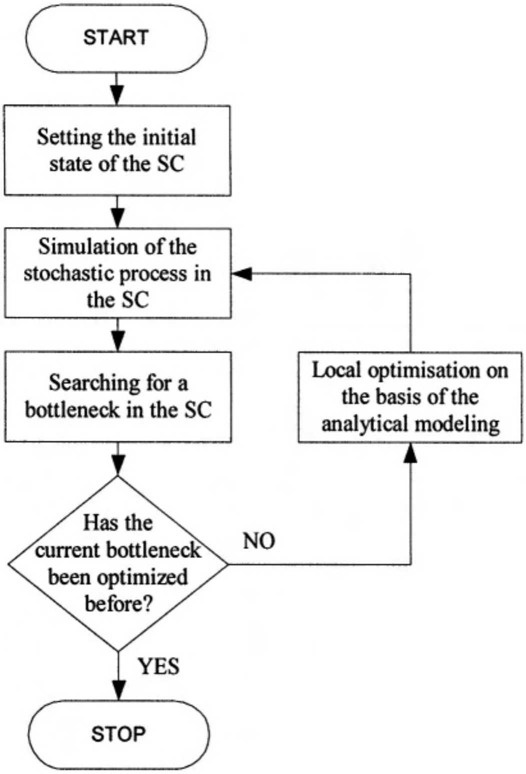

START
Setting the initial state of the SC
Simulation of the stochastic process in the SC
Searching for a bottleneck in the SC
Local optimisation on the basis of the analytical modeling
Has the current bottleneck been optimized before?
NO
YES
STOP

Figure 8-2. SC optimisation algorithm

For a general SC structure and a general kind of arrival and servicing processes analytical methods using queuing models cannot be used. Therefore, in the general case we use a simulation model.

The next step of the algorithm is the bottleneck's extension, which can be done by using an analytical method. We minimise the value of the criterion function for the servicing node being the bottleneck using the analytical method and the Kleinrock's approximation [5].

The stages of the algorithm are explained below.

1. Setting the initial state of the model

We can consider the processes incoming into each servicing node as independent ones. This gives us the opportunity to set up the initial values of the decision variables for each servicing node independently with the following assumptions: the number of servers for each servicing node is set by using the condition of balance between rates of the incoming workflow process and productivity of the servicing node [5].

$$P_{i0} = \frac{1}{\mu_i} \sum_k r_{ki} \lambda_k v_k, \quad k = 1,\ldots,K, \quad i = 1,\ldots,I.$$

2. Simulation of the stochastic process in the SC

Great possibilities for analysing SC and making an optimal decision are provided by the simulation [1], which has no restrictions on the dimension, the kind of arrival pattern, the discipline and the time of servicing. An examined SC represents an open queuing network and can be modelled as a set of tandem queues [2].

The simulation model was constructed under the following conditions:

a) The SC has several kinds of workflow processes and consists of a number of servicing nodes.

b)Each servicing node represents a multi-channel queuing system equipped with identical servers. All servers, operating in parallel, have the same productivity, work-time and idle-time cost.

c)Incoming streams of jobs have different parameters of arriving and servicing (distribution laws, rates of arrival and servicing).

d)The capacities of the input buffers at servicing nodes are unlimited.

e)The discipline of servicing is 'First come-First served'.

3. Searching for a bottleneck in the SC

The simulation model allows defining the following variables for each servicing node:

– the average system flow time of a work,

– the utilisation and average idle time of servers.

The value of the criterion function can be calculated for each node n_i basing on the mentioned variables for each servicing node. A 'bottleneck' with the maximal value of the first component C_I^i of the criterion function can be selected from all servicing nodes.

4. Local optimisation on the basis of an analytical modelling

The goal of this stage is the optimisation of the decision variable P_i of the 'bottleneck' servicing node, providing the minimal value of CR_i. Such optimisation can be quickly carried out on the basis of analytical modelling:

a) Kleinrock's approximation

Analytical modelling of queuing system is a very fast and robust technique for solving the optimisation problem. Unfortunately we deal with a system with many input streams and a general kind of arrival and servicing processes. For such systems analytical optimisation is very difficult or even unsolvable. According to the Kleinrock's approximation [5] the sum of all arriving process in a servicing node can be modelled as a Markovian one if the correlation between the different processes, arriving at the queuing system, is not present. Merging several processes at the input of a queuing system as shown has an effect similar to restoring the independence of inter-arrival times and job service time. Therefore, using the Kleinrock independence approximation and assuming that a total arriving stream is Poisson, we have:

$$\Lambda_i = \sum_{k=1}^{K} \lambda_k r_{ki}, \quad M_i = \frac{\Lambda_i}{\tau_i \sum_k \lambda_k r_{ki} v_k},$$

where Λ_i is the total arrival rate, M_i is the average service rate in servicing node n_i.

b) Markovian system *M/M/m*

Using Kleinrock approximation *G/G/m* queuing system can be modelled as *M/M/m* system. For that kind of system all necessary formulas can be derived and literature presents comprehensive set of equations describing the *M/M/m* system [5, 7].

c) Local optimisation

Minimisation of the local criterion function for a selected servicing node is the discrete optimisation problem. Basing on the fact that criterion function CR_i is convex (Figure 8-3) and has only one variable P_i we can design simple optimisation algorithm. Starting form the initial value of P_i we can increment it each time checking the condition $CR_i(P_i) < CR_i(P_i+1)$. When this condition is fulfilled it means that the minimum value of the criterion function CR_i is reached and optimal number (local optimum) of parallel servers in servicing node equals P_i.

This algorithm is presented in Figure 8-4.

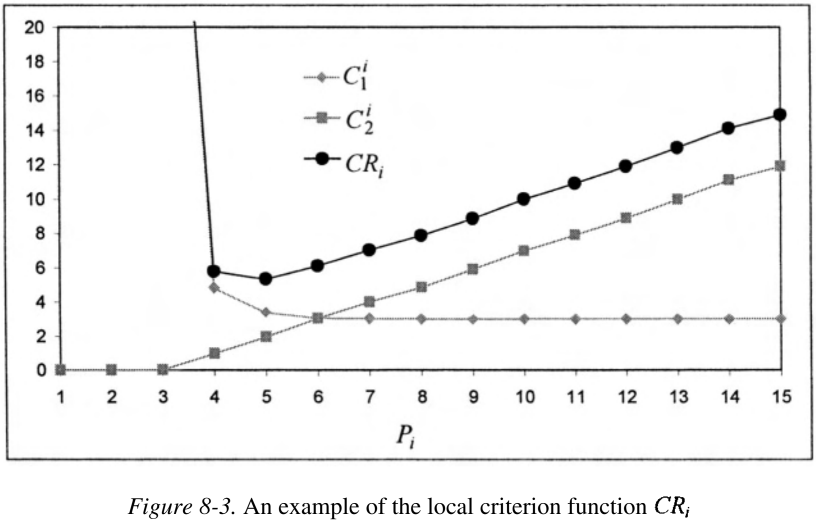


Figure 8-3. An example of the local criterion function CR_i

In Figure 8-3 C_1^i is the work in progress cost for node n_i, C_2^i is the cost of servers' utilisation for node n_i.

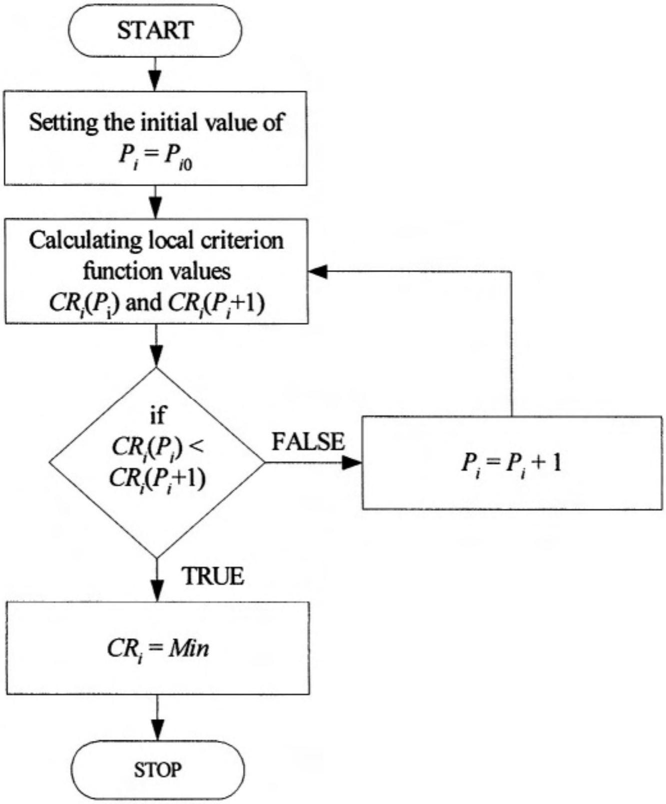

START
Setting the initial value of $P_i = P_{i0}$
Calculating local criterion function values $CR_i(P_i)$ and $CR_i(P_i+1)$
if $CR_i(P_i) <$ $CR_i(P_i+1)$
FALSE
$P_i = P_i + 1$
TRUE
$CR_i = Min$
STOP

Figure 8-4. Local optimisation algorithm

5. Stop condition

Decision about continuing the optimisation process should be made at each iteration of the SC optimisation algorithm. In order to stop the algorithm it has to be determined if further optimisation will improve a resource allocation in the SC and minimise the value ofthe criterion function *CR* (1). It can be done be simply checking whether currently found servicing node n_i being 'bottleneck' was optimised at the previous iteration.

4. A NUMERICAL EXAMPLE

A typical example of a publishing SC is a prepress centre, which includes a set of specialised workplaces: a flatbed or drum scanners, a graphic station for image processing, a raster image processor, a image setter, and a colour proofing unit. The incoming workflow of customer's demands corresponds to a certain kind of publication, characterised by the kind of information (textual, graphic, illustrative) and the set of technical parameters (format, colours, screen line number). At the prepress centre the special technological route of the servicing node is organised for each kind of publications. Each servicing node has several equipment units of a certain kind. For example, the flatbed and drum-type scanners can be incorporated in one servicing node as the interchangeable equipment.

The technological graph for the prepress stage of the publishing process is presented in Figure 8-5. The nodes 1, 2,.., 13 are explained below.

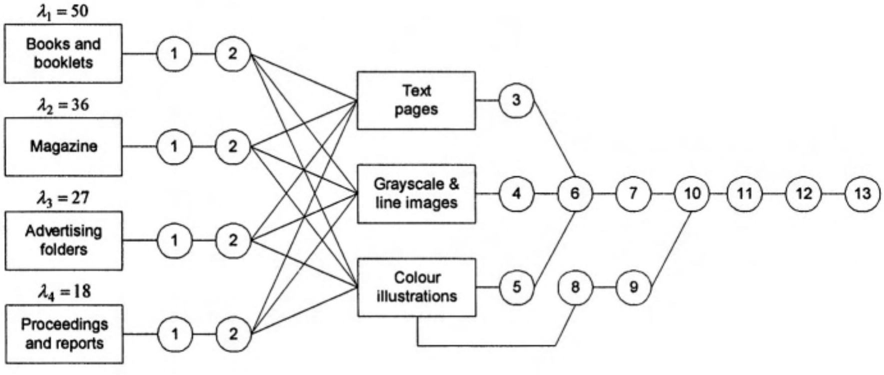


Figure 8-5. Technological graph of the prepress process

As it is shown in the figure there are several workflows in the publishing process, corresponding to different kinds of publications. There are four kinds of publications from the point of view of the arrival pattern:

- books and booklets, performed accordingly to the author's demands, which times of arrival and servicing are characterised by Normal distribution,
- occasional printing works, e.g. advertising, folders, booklets, etc., which are characterised by random times of arrival and servicing,
- periodical press, e.g. bibliographic, academic periodical publications, entertainment and commercial magazines, etc., which are characterised by deterministic times of arrival and servicing,

- internal publications, e.g. proceedings and reports, which are characterised by the general time of arrival and servicing.

We know volume (size) distribution law and specification of each kind of publication, given in average number of each kind of page (see Table 8-1).

Table 8-1. Specification of publication

Type of publication	Average number of pages			Distribution law
	text pages	greyscale & line images	Colour illustrations	
1. *Books and booklets*	180	8	2	Normal
2. *Magazine*	25	27	40	Deterministic
3. *Advertising folders*	8	2	50	Poisson
4. *Annual Reports*	20	20	15	Exponential

Some characteristics of the technological operations for the prepress process, presented in Figure 8-5, are shown in Table 8-2.

Table 8-2. Characteristics of technological operations

N	Type of operation	Operational time (min. per page)
1	Project's design within the client's budget	10*
2	Publication design	40*
3	Text edition	10
4	White and black images scanning	1
5	Colour images scanning	4
6	Composition	2
7	Design proof	1
8	Image-processing (Colour correction, colour separation)	20
9	Colour image proofing	5
10	High-quality camera-ready art, suitable for negative-plate production Imposition of publication	10
11	Image processing	5
12	Film Negative Making	15
13	Plate Making	1

* Time for a whole publication

For selected tests, simulation experiments were conducted under the following conditions:

1. The simulation model is realised for four independent streams of input flows:

– a Poisson law ofjobs arrival for books and booklets
– a deterministic law ofjobs arrival for magazines
– a Normal law ofjobs arrival for advertising folders
– a deterministic law ofjobs arrival for annual reports

2. Servicing time is fixed (deterministic) for each kind of page, but number of pages in a publication is a random variable (see Table 8-1).
3. The replication length is 90 days i.e. 2160 hours.

For the initial configuration of the system (obtained using the condition of balance), results are shown in the Table 8-3.

Table 8-3. Values of the criterion function for all servicing nodes, $\alpha = 0.12$, $\beta = 45$

node n_i	Number of servers P_i	Flow time [h] $(\tilde{\tau}_i^W + \tau_i^S)$	Utilisation ρ_i	WIP cost C_1^i	Cost of server util. C_2^i	Value of criterion function CR_i
1	1	1.60	0.3194	8.19	42.03	46.82
2	1	4.52	0.605	23.14	24.75	45.91
3	1	0.84	0.214	4.30	41.30	41.67
4	4	21.65	0.7573	74.46	68.54	138.14
5	1	0.31	0.0969	0.15	50.16	45.79
6	1	2.50	0.5175	3.08	29.13	29.80
7	2	1.85	0.5267	9.47	57.33	62.07
8	1	1.92	0.5264	9.83	28.68	36.14
9	4	8.88	0.7055	10.95	78.90	83.96
10	1	4.55	0.7042	5.61	19.79	23.92
11	**3**	**17.18**	**0.8662**	**87.95**	**65.07**	**151.01**
12	2	1.52	0.5179	7.78	78.21	81.17
13	1	2.53	0.6287	12.95	38.57	49.66

Total cost of 'work in progress': $C_1 = 258$

Total cost of 'servers utilisation': $C_2 = 622$

Total value of criterion function: $CR = C_1 + C_2 = 880$

As shown in Table 8-3 the servicing node n_i = 11 was found as the bottleneck. The direct cause of it is the high level of servers utilisation and long time of orders 'flow' through the servicing node. The value of the "work in progress" cost for this servicing node is the highest one and it's equal to 87,95. The value of criterion function is equals to 153.

Using the analytical procedure developed by Zaikin *et al.* [7] the optimal number of servers for the eleventh servicing node was found. Table 8-4 presents the value of the criterion function depending on the number of servers working in parallel in the eleventh servicing node under condition that no other parameter has been changed. The conducted experiments (Table 8-4) showed that the optimal number of servers for Film Negative Making (11th operation) is 4.

This was only the first iteration of the developed algorithm. Afterwards the next bottleneck could be searched for and its optimal parameters could be analytically found using again the results from [10].

Table 8-4. A comparison of different configurations of 11th servicing node

node n_{11}	Number of servers P_{11}	Flow time [h] $(\tilde{\tau}_i^W + \tau_i^S)$	Utilisation ρ_{11}	WIP cost C_1^{11}	Cost of serv. util. C_2^{11}	CR_{11}	Total value of CR
11	3	17.18	0.8662	87.95	65.07	151.01	836.07
11	**4**	**4.1221**	**0.6374**	**21.10**	**125.27**	**146.37**	**807.33**
11	5	3.2546	0.542	16.66	189.50	194.71	878.13

After the second bottleneck searching it proved that the node with the highest value of CR_i was again the node number 11. According to the stop condition described in Section 3 the SC optimisation algorithm can be stopped. The results of the second bottleneck searching are shown in Table 8-5.

The Algorithm can be improved if the bottleneck searching is continued excluding the node currently optimised. However, the stop condition presented in Section 3 is sufficient to satisfactory improve the SC.

Table 8-5. Final solution for $\alpha = 0.12$, $\beta = 45$

Node n_i	Number of servers P_i	Flow time [h] $(\tilde{\tau}_i^W + \tau_i^S)$	Utilisation ρ_i	WIP cost C_1^i	Cost of servers utilisation C_2^i	Local criterion function CR_i
1	1	1.5068	0.3261	7.71	38.33	46.04
2	1	4.7831	0.6347	24.49	21.44	45.92
3	1	0.8014	0.2047	4.10	37.79	41.89
4	4	14.3232	0.7058	49.26	72.96	122.22
5	1	0.3224	0.0972	0.16	45.63	45.78
6	1	2.8789	0.5569	3.55	24.94	28.49
7	2	1.7485	0.5159	8.95	53.57	62.52
8	1	1.7309	0.5145	8.86	26.85	35.71
9	4	10.4122	0.7569	12.84	63.76	76.60
10	1	6.6162	0.7495	8.16	16.27	24.43
11	**4**	**4.1221**	**0.6374**	**21.10**	**125.27**	**146.37**
12	2	1.5546	0.507	7.96	74.37	82.33
13	1	2.3715	0.6248	12.14	36.88	49.02

Total cost of 'work in progress': $C_1 = 169$

Total cost of 'servers utilisation': $C_2 = 638$

Total value of criterion function: $CR = C_1 + C_2 = 807$

5. CONCLUSION

The problem of the resources' allocation in publishing supply chain is formulated as a task of optimisation of a queuing model parameters. The objective function depends on the stochastic and deterministic variables, such as distribution law of arriving orders, distribution law of servicing time, number of parallel servers for each node of the queuing system, etc.

Developed approach based on coupling analytical and simulation models. Bottleneck nodes are searched by simulation, and analytical method is used to local optimisation of current bottleneck node.

ACKNOWLEDGEMENTS

This research was supported by the Polish-French program ‘Polonium’. The authors would like to thank Prof. A. Dolgui for the discussions, helpful ideas and comments.

REFERENCES

1. Azadivar F., and Lee Y., 1988. Optimisation of discrete variable stochastic systems by computer simulation, *Mathematics and Computers in Simulation,* 2, 331-345.
2. Bertsekas D., and Gallager R., 1992. *Data Networks,* Prentice-Hall, Englewood Cliffs NJ
3. Dolgui A., Zaikin O., Kushtina E., Korytkowski P., 2003. Modelling and optimization of the processing nodes performance in Computer aided Control Systems of distributed Production of Printer Matter. *Automation and Remote Control,* 64 (9), 1501 – 1506.
4. Kelton W.D., Sadowski R.P., Sadowski D.A., 1998. *Simulation with Arena,* McGraw-Hill, Boston.
5. Kleinrock L., 1976. *Queueing Systems,* John Wiley & Sons, N. Y.
6. O’Quin D., 2000. *Print Publishing: A Hayden Shop Manual,* Haiden books, Indiana.
7. Zaikin O., 2002. Resource Distribution in Automatic Production Control for Nonmaterial Products: A Mathematical Model, *Automation and Remote Control* 63 (8), 1351-1356.
8. Zaikin O., Dolgui A., Korytkowski P., 2002. Optimisation of Resource Allocation in Distributed Production Networks, In: *Lecture Notes in Artificial Intelligence,* 2296, Dunin-Klepicz B., Nawarecki E. (Eds.), Springer-Verlag, Berlin, 322-332.
9. Zaikin O., Dolgui A., Korytkowski P., 2002. Modeling and performance optimization of a distributed production system. *Information Control Problems in Manufacturing* 2001. *A Proceedings volume from the* 10*th IFAC Symposium,* P. Kopacek, G. Morel, M.B. Zaremba (Eds.), Elsevier Science, 105-110.
10. Zaikin O., Kushtina E., Woloshyn M., Korytkowski P., 2001. Stochastic simulation modeling in manufacturing resources planning for publishing enterprise, In: *Advanced Computer Systems,* Soldek J., Pejas J. (Eds.), Informa, Szczecin, 167-175.

PART II:OPTIMISATION METHODS

This page intentionally left blank

Chapter 9

HYBRID METHODS FOR LINE BALANCING PROBLEMS

Corinne Boutevin, Laurent Deroussi, Michel Gourgand, Sylvie Norre

Abstract: A line balancing problem is defined by a line along which vehicles go througth and are progressivly assembled. The assembly operations are performed by workstations spread along the line. The objective is to assign operations to workstations in order to minimize, for instance the number of required workstations. The basic constraints are cycle time and precedence constraints. To solve this problem, we have firstly used a genetic algorithm with different operators (some of them have been proposed in the literature). We suggest to couple this genetic algorithm with some heuristics (which have been previously published). We then obtain hybrid methods that improve the obtained offsprings, before inserting them in the population. We have tested these methods on literature instances (one range of vehicles and cycle time and precedence constraints), and on generated and industrial data. These real instances represent a real problem in the automotive industry.

Key words: line balancing, genetic algorithms, hybrid methods.

1. INTRODUCTION

The assembly line balancing problem (ALBP) is commonly met in the industry. This problem concerns the assembly of any kind of objects. So it appears in microelectronic chips or electronic household appliances assembly and of course in the automobile industry. It consists in progressively assembling an object while it is going through the assembly line (with a constant speed). The goal of the line balancing problem is to spread the assembly operations along the line, more precisely, among workstations placed on the line. Two problems can be considered: the first one consists in fixing the cycle time and minimizing the number of workstations and the second one consists in fixing the number of workstations and minimizing the cycle time. We consider the first situation. To solve this problem, we propose to use genetic algorithms combined with heuristics.

In Section 2, we describe our industrial problem. In Section 3, we present general principle of genetic algorithm and several literature operators. In Section 4, we describe our proposed genetic algorithm, and in Section 5, hybrid methods. In Section 6, we show computational results obtained on literature instances, on generated and industrial data.

2. DESCRIPTION OF THE PROBLEM

Our balancing problem is an industrial problem (collaboration with a French automobile industry). The vehicles go through the line and are progressively assembled. The operations are performed by workstations. The line is divided in sections which contain one or more workstations (which perform, at the same time, operations on the same vehicle).

An operation requires a storage area for parts and tools used for its performance ([3]). Required operations depend on the range of the vehicle (we consider several ranges of vehicles; it is a mixed-model assembly line problem). A vehicle is available on a section during a limited duration; so required operations have to be performed during this time. Figure 9-1 describes the line.

Our balancing problem is an industrial problem with standard and specific constraints.

Standard constraints are:

- **Constraints C1 on the cycle time:**
  The vehicle is accessible by workstations placed on a given section during a limited duration called the cycle time. For each workstation *j* (parallel stations) and each range *m* of vehicles (the required operations depend on the range of the vehicle), the sum of processing times of all operations required by *m* and performed by *j* must be inferior to the cycle time.

Specific constraints are:

- **Constraints C2 on the section length:**
  The performance of an operation requires an area to store involved parts and tools. For each section *k,* the sum of storage area required by all operations performed on *k* must be inferior to the length available on the section *k,*
- **Constraints C3 on the operator time:**
  The number of considered vehicles (sequence of vehicles in input of the line) corresponds to the load of a given period (daily period). To assemble

this quantity of vehicles, the operator has a limited duration called the operator time. For each workstation *j*, the sum of allocated times (1.2 times the processing times) of operations performed by *j* and required by all vehicles of the sequence must be inferior to the operator time,

- **Constraints C4 on incompatibility between operations:**
  Incompatible operations can not be assigned to the same workstation,
- **Constraints C5 on precedence between operations:**
  If operations i_1 and i_2 are linked by a precedence relation, it means that i_1 and i_2 must be assigned to the same workstation (as precedence constraints defined in the literature) or i_1 must be assigned to a workstation placed on a section located before (on the line) the one where i_2 is performed.

The objective function is the minimization of the number of used workstations (a used workstation performs one or more operations).

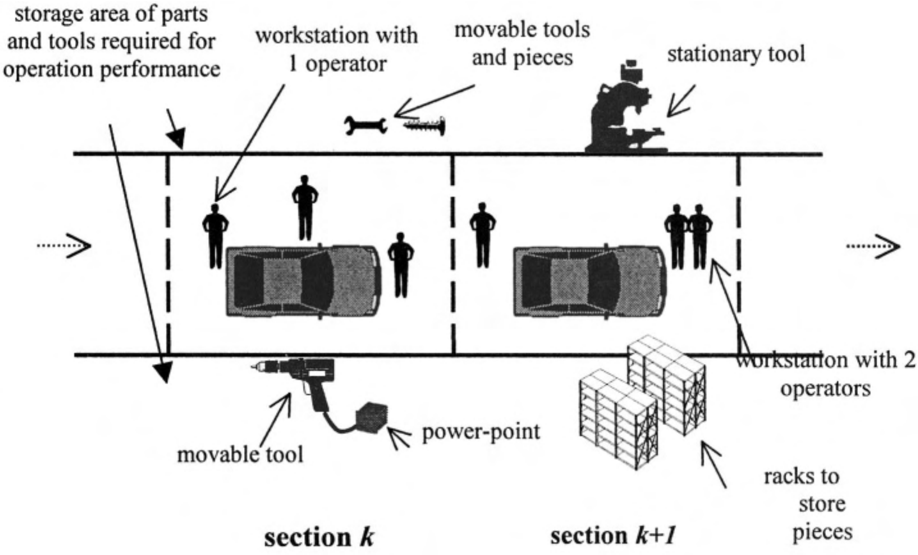

storage area of parts and tools required for operation performance
workstation with 1 operator
movable tools and pieces
stationary tool
workstation with 2 operators
movable tool
power-point
racks to store pieces
section *k*
section *k+1*

Figure 9-1. Description of the line

3. GENETIC ALGORITHMS: PRINCIPLE AND STATE OF THE ART

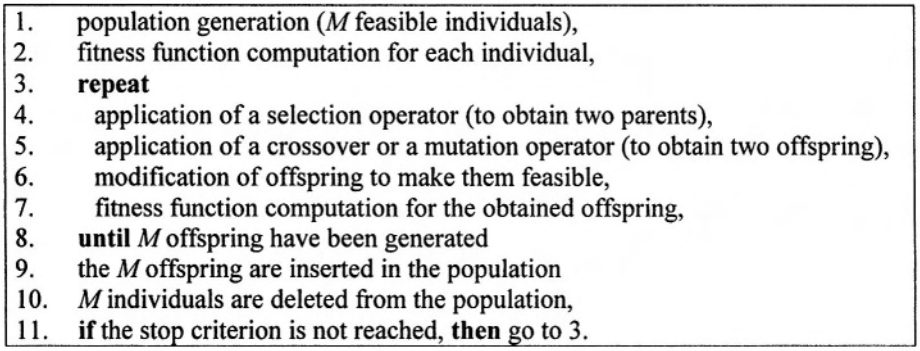
3.1 General principle

This evolutionist method has been introduced by [7]. It considers a population of individuals; each individual represents a feasible solution for the studied problem. An individual is schematized by a chromosome constituted by many genes. An allele is a gene value and the locus is the gene position into the chromosome.

An individual evaluation (quality evaluation) is made by the fitness function, which can be:

- identical to the objective function (minimization of the number of used workstations in the studied case),
- different from the objective function. But the fitness function has to evolve in the same way as the objective function (indirectly optimizes the objective function).

At each iteration, parents are selected (selection) and offspring are generated (crossover and mutation). The parent selection and the offspring generation can be different; one or more children are created, which are based on two or more parents. The offspring are inserted (insertion) and a sub-set of individuals are remained (it allows to have a constant number of individuals in the population). So the population evolutes during the algorithm (via several operators), in order to lead to the optimum. Figure 9-2 shows the general scheme of a genetic algorithm.



```
1.  population generation ($M$ feasible individuals),
2.  fitness function computation for each individual,
3.  **repeat**
4.    application of a selection operator (to obtain two parents),
5.    application of a crossover or a mutation operator (to obtain two offspring),
6.    modification of offspring to make them feasible,
7.    fitness function computation for the obtained offspring,
8.  **until** $M$ offspring have been generated
9.  the $M$ offspring are inserted in the population
10. $M$ individuals are deleted from the population,
11. **if** the stop criterion is not reached, **then** go to 3.
```

Figure 9-2. Genetic algorithm scheme

3.2 State of the art

Genetic algorithms are sometimes used to solve the line balancing problem and provide good results. But, we notice that these methods are scarcely quoted in surveys (for instance [12]).

In [9], many ways to code a solution by a chromosome are summarized:

- *station-oriented representation:*

  if the operation i is assigned to the station j, then the station j is placed in i^{th} position in the chromosome (the alleles give the station numbers) ([1], [6]). Figure 9-3 presents a precedence graph and the operation assignments. The associated chromosome is: 2 2 1 1 1 2 3 4 3 4

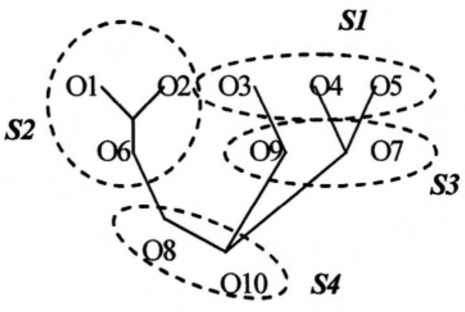


Figure 9-3. Genetic algorithm scheme

– *sequence-oriented representation:*
operations are listed in the order they are assigned to stations (the alleles give operations). The gene order in the chromosome gives the operation performance order. The representation of Figure 9-3 can be:
3 4 5 1 2 6 9 7 8 1 0

– *partition-oriented representation:*
it is based on the sequence oriented representation, in which are added separators. These separators allow partitioning the operations into several stations ([11]). This representation does not give the sequence of operation performance inside each station. We have
3 4 5 | 1 2 6 | 9 7 | 8 10

In [1], a genetic algorithm is proposed for the line balancing problem. The authors use a station-oriented representation. The parent selection is made by associating an interval to each individual (whose length is proportional to the individual quality) and by randomly selecting two intervals. The used crossover is a one crossing point which consists in randomly choosing one crossing point r in [1, *length*], and in exchanging parents' genes placed at the right of r, in order to obtain two offspring. The mutation consists in modifying some gene values (assignments). Finally, the fitness function integrates penalties for constraint violations (the stage 6 of Figure 9-2 is not applied).

In [5], genetic algorithms are applied for different optimization problems (bin packing, traveling salesman problem, multiprocessor scheduling). For the bin packing problem (which is a line balancing problem without precedence constraints [6]), a partition-oriented representation is used. They have tested one and two crossing point crossovers. The two crossing point operator creates two children, e_1 and e_2, based on two parents p_1 and p_2, by using two crossing points r_1 and r_2 $(r_1 \neq r_2)$ randomly chosen in [1, *length*], where *length* corresponds to the number of genes contained in the chromosomes. The two children are obtained by reverse the central parts of parents (genes placed between r_1 et r_2). The used mutation consists in permuting two genes. The fitness function is the value returned by the First Fit heuristic applying to the individual. Offspring replace the worst individuals, if they are better than them.

In [11], a partition-oriented genetic algorithm for the line balancing problem is proposed. The initial population is randomly generated and two parents are randomly selected in each iteration. The mutation operator permutes two randomly chosen genes, which are placed in distinct parts (we notice that the number of used stations is unchanged, because the considered objective is the minimization of the cycle time for a given number of stations). Before inserting offspring in the population, a reordering is made to obtain their feasibility (if necessary). Then the two generated offspring replace the worst individuals.

Original algorithms have been proposed in [6] and [10]: the grouping genetic algorithms. This method is based on the idea that the line balancing problem is a grouping problem. A grouping problem is a kind of problems where the goal is to group a set of objects (operations) into a small number of families (workstations). The objective is the minimization of the number of required families (submitted to many constraints). The coding scheme and the operators are adapted to the notion of operation grouping. The method is group-oriented and not object-oriented (as the other methods). The authors show that the information that a workstation is empty is more important than operation assignments. They propose chromosome coding scheme with a "standard" part and a grouping part. This last part contains all used stations. The standard part gives the operation assignment (station-oriented representation). We notice that the chromosome length varies during the algorithm. Operators act only on the grouping part. So, as the method is group-oriented, operators are made to directly modify groups (workstations) and not the operations. At each iteration, only one child is created with a two crossing points procedure. The genes placed between the two points, in the second parent, are inserted, in the first parent, after the first crossing point.

Example: A | BCD | EF (first parent's grouping part)
AB | *CD* | (second parent's grouping part)

If we have $r_1 = 1$ and $r_2 = 4$ in the first parent, and $r_1 = 2$ and $r_2 = 4$ in the second parent, the generated child's grouping part is: A*CD*BCDEF. If all operations contained in *C* and *D* are also contained in C, E and F, we then delete C, E and F. [10] proposes to add all deleted operations (contained in C, E or F and not present in *C* and *D*) with the First Fit Decreasing heuristic. The second child is obtained in the same way.

Table 9-1 summarizes all previously presented characteristics. We notice that it is difficult to implement their method due to the lack of details and that all algorithms have been tested only on generated instances.

Table 9-1. Previously presented characteristics

	[1]	[5]	[11]	[6], [10]
Genetic algorithm scheme				
population generation			X	
fitness function	X	X		X
selection operator	X		X	
crossover operator	X	X		X
mutation operator	X	X	X	X
offspring feasibility			X	X
insertion operator		X	X	
Tested instances				
literature				
real-case				
generated	X	X	X	X

4. PROPOSED GENETIC ALGORITHM

In this section, we present a genetic algorithm, in which we have implemented different operators. The general scheme of this method is given in Figure 9-4.

We propose to use a station-oriented representation. The sequence-oriented representation can not be applied because of the existence of parallel workstations (all workstations placed in the same section can be assimilated to parallel stations). The first parent is described in Figure 9-3. Table 9-2 gives operation characteristics.

We consider 10 vehicles from the same range and one operator per workstation. The cycle time and the operator time are respectively equal to 10 and 140. Workstations S1 and S2 are placed on the section 1, S3 is on the section 2 and S4 on the section 3. The length of S1 is equal to 30, it is equal to 15 for S2, and the length of S3 is 10. Incompatibilities are between O1 and O7, O2 and O5, O6 and O10.

1. Population generation (feasible individuals),
2. fitness function computation for each individual,
3. application of a selection operator,
4. application of a crossover operator,
5. application of a mutation operator,
6. modification of offspring to make them feasible,
7. fitness function computation for the obtained offspring,
8. application of an insertion operator,
9. **if** the stop criterion is not reached, **then** go to 3.

Figure 9-4. Our genetic algorithm scheme

Table 9-2. Instance characteristics

operation	01	02	03	04	05	06	07	08	09	010
processing time	1	3	3	2	4	4	6	2	2	2
storage area	3	3	6	6	5	5	2	2	1	3
allocated time	1.2	3.6	3.6	2.4	4.8	4.8	7.2	2.4	2.4	2.4

4.1 Population generation

As it is shown in Table 9-1, the generation of the initial population is not usually described in the literature. In our genetic algorithm, we consider a population of 50 individuals. This initial population computation is made by using an algorithm based on the COMSOAL heuristic ([2]). This heuristic consists in successively loading each workstation. We consider a set O of operations and a priority list L on workstations:

- the set O contains all the feasible operations (operations not yet assigned and whose all predecessors have been already assigned); this set is computed after each assignment,
- in the priority list L, workstations placed at the beginning of the line have the highest priorities.

Workstations are considered according to the decreasing priority in L. An operation is randomly chosen in the set O and is assigned to the current workstation. If this assignment is unfeasible, the next workstation in the list L is considered; it is the new current workstation. If the assignment is feasible, the operation is assigned to the current workstation, is deleted from O and the set O is updated. Another operation is randomly chosen in the set O, and so on. In this way, we generate the initial population of feasible chromosomes.

4.2 Fitness function

The objective is the minimization of the number of used workstations. The fitness function must indirectly improve this objective (or can be identical to the objective). We propose to use a hierarchical fitness function based on two criteria:

- $g_1 = K'$ $(g_1 \geq 1)$, where K' is the number of used workstations,
- g_2 is a fraction depending on the sum of idle times. This fraction is given by (1), where:
  - CT is the cycle time,
  - K is the number of considered workstations on the line,
  - ld_j is the maximal load of the workstation j (for each range of vehicles, we compute the sum of processing times of operations required by the range and assigned to the workstationy j and we keep the greater one),
  - OP_j is the number of operators using the workstation j,
  - N is the operation number.

$$g_2 = \frac{\sum_{j=1}^{K} ld_j \times Min\,(1, CT \times OP_j - ld_j)}{N.\sum_{j=1}^{K} (CT \times OP_j - ld_j)^2} \qquad (0 \le g_2 \le 1) \qquad (1)$$

g_2 allows to not privilege two workstations equally filled, but to accept one workstation nearly full and another one nearly empty. This last situation allows making a workstation empty. And the hierarchical fitness function to minimize is defined by $f = g_1 + g_2$. As g_1 returns natural values ($g_1 \ge 1$) and $0 \le g_2 \le 1$ (g_1 represents the whole part of f and g_2 its decimal part), the minimization firstly concerns the function g_1 and secondly g_2

4.3 Selection operator

We have used the selection operator proposed in [1] which consists in associating an interval to each individual. In the case of a maximization (respectively minimization), interval length is proportional (respectively inversely proportional) to the fitness value f of the individual. For a minimization, the interval length for the individual c is defined by (2).

$$l(c) = A \times \frac{1}{f(c)} \times \left[\sum_{t=1}^{M} \frac{1}{f(t)}\right]^{-1} \qquad (2)$$

where A is a constant (depending on the fitness function) allowing choosing more or less good individuals according to its value. Indeed intervals are normalized on [0, K] and are juxtaposed on [0, K]. Then we uniformly chose two random numbers a_1 and a_2 (one for each parent) on [0, K]. The parent p_1 (respectively p_2) is the individual c whose the interval l(c) contains a_1 (respectively a_2). We can formalize it with the formula (3):

$$p_1 = \min \left(p,\, p \in [1, M] \; / \; a_1 \ge \sum_{c=1}^{p} l(c) \right) \qquad (3)$$

Example: let be a population of 8 individuals. Each individual has its interval normalized on [0, K], where K is equal to 200. Figure 9-5 shows intervals successively placed on [0,200]. If a_1 is equal to 50, it is contained on the interval associated to the third individual; so p_1 represents the individual 4. In the same way, if a_2 is equal to 125, p_2 is the fifth individual.

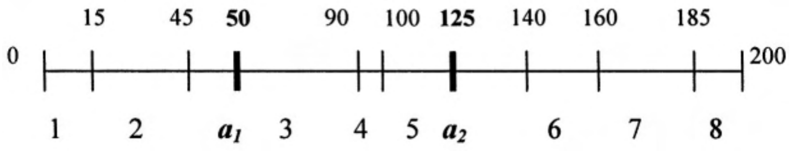


Figure 9-5. Illustration of parent selection

We have chosen this operator because it does not automatically select the best individuals but only good quality ones. It allows remaining population diversification. Indeed, the best individual selection operator does not allow ensuring diversification (selected chromosomes represent near solutions and so will generate offspring which are near from the parents).

The chromosome associated to the first parent is:

p_1 = 2 2 1 1 1 2 3 4 3 4 (cf. Figure 9-3).

We suppose that the second parent is:

p_2 =1 1 2 2 3 1 3 1 2 4.

4.4 Crossover operator

We propose to use a two crossing points operator because it allows a diversification but not perturbs a lot the population.

By considering r_1 = 4 and r_2 = **8**, the two obtained offspring are:

e_1 = 2 2 1 1 3 1 3 1 3 4

e_2 = 1 1 2 2 1 2 3 4 2 4

4.5 Mutation operator

The proposed mutation operator consists in computing, for each child, the less loaded workstation (according to C1 constraints) and to modify all genes equal to it (the new assignments are randomly computed). Indeed, as the objective is the minimization of the number of used workstations, this mutation operator allows making a workstation empty.

In both previous offspring, the workstation 4 is the less loaded one. So they become, after applying this mutation operator:

e_1 = 2 2 1 1 3 1 3 1 3 3

e_2 = 1 1 2 2 1 2 3 2 2 3.

4.6 Offspring feasibility

The insertion operator can be applied only after that offspring have been made feasible (if necessary), as shown in Figure 9-2. A child can be unfeasible due to the application of the crossover or mutation operators. Offspring are unfeasible if an operation is assigned to two stations, if it is not assigned, or if constraints are not satisfied... The method used to make offspring feasible depends on the coding scheme.

To make offspring feasible, we determine unfeasible workstations and sections. We then randomly choose, as necessary, an operation assigned to it and we move this operation to another workstation. If any workstation can perform the operation, a new workstation is used.

We notice that offspring obtained after the crossover and the mutation are not feasible (in e_1, the workstations 1 and 3 do not satisfy the cycle time constraints, and in e_2, the workstation 2 does not satisfy these same constraints). An example of feasible offspring is:

$e_1 = 2\ 2\ 1\ 1\ 1\ 2\ 3\ 2\ 3\ 3$

$e_2 = 2\ 2\ 1\ 1\ 3\ 2\ 3\ 2\ 1\ 4$

4.7 Insertion operator

The insertion operator inserts the offspring, in the population, if they are better than their two parents; they replace the worst parent always in the population. This operator allows keeping population diversity. Indeed, if offspring replace the worst individuals (and not their parents), all the population will tend to near solutions.

The fitness function for e_1 is equal to 3.9, it is equal to 4.012 for e_2, 4.081 for p_1, 4.012 for p_2. So e_1 replaces p_1 (the worst parent) and e_2 is not inserted (p_2 remains).

5. HYBRID METHODS

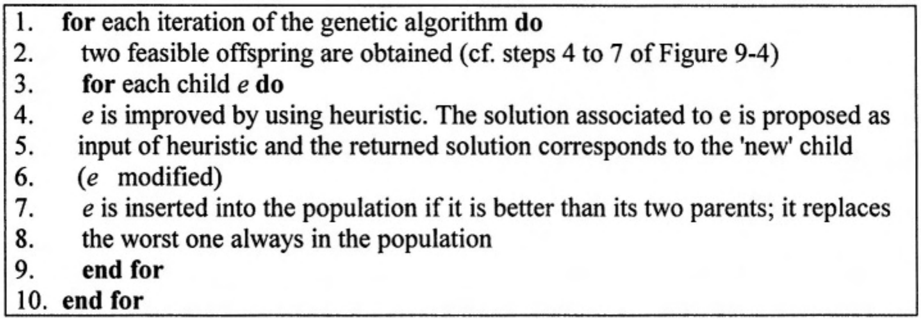
In the previous section, we have only presented the genetic algorithm. This method can allow obtaining good results. But it can be improved by combining it with other optimization method(s). These combinations are named *hybrid methods.*

We propose to create hybrids methods by combinations of the previously presented genetic algorithm with our heuristics: FRO, MLB1 and MLB2. These heuristics have been detailed in [4]. We briefly present them:

- *MLB1* (Maximum Load-Based) is a load-based heuristic. This method unloads under-loaded workstations or sections and loads the most loaded workstations and sections,
- *MLB2* is a variant of MLB1. It consists of not separately considering operations but moving groups of operations. All operations assigned to the same workstation are moved to other workstations,

- *FRO* (First Realized Operations) is a priority-based heuristic. The priority rule is based on the initial balancing proposed as input solution for the heuristic (a given operation scheduling). Operations with the highest priority are the ones assigned to workstations placed in the beginning of the line. Workstations are considered one by one and operations assigned to them are chosen according to their priority and if no constraint is violated.

The combination between the genetic algorithm (GA) and a heuristic is noted by GA ↔ ***heuristic***. The description of this combination is given in Figure 9-6.



```
for each iteration of the genetic algorithm do
    two feasible offspring are obtained (cf. steps 4 to 7 of Figure 9-4)
    for each child e do
        e is improved by using heuristic. The solution associated to e is proposed as input of heuristic and the returned solution corresponds to the 'new' child (e modified)
        e is inserted into the population if it is better than its two parents; it replaces the worst one always in the population
    end for
end for
```

Figure 9-6. "GA ↔ heuristic" description

Tested combinations are: *GA ↔ FRO*, *GA ↔* (*MLB1*+*MLB2*) and *GA ↔* (*MLB2*+*FRO*). The notation *'heur1+heur2'* means that *heur1* is chained with *heur2;* the result returned by *heur1* is proposed as input (initial solution) of *heur2*. The next part shows that these hybrid methods really improve the effectiveness of the genetic algorithm and of the heuristics.

6. COMPUTATIONAL RESULTS

Proposed hybrid methods have been tested on literature instances[1]. We have used 50000 iterations for the genetic algorithm. These 269 instances concern one range of vehicles, from 5 to 297 operations and take into account only cycle time and precedence constraints. Obtained results, for the fitness function *f*, are summarized in Table 9-3. We have used the fitness function *f*(described in 4.2). We present the results obtained with the genetic algorithm *(GA),* the heuristic *FRO,* the combinations and optimal results computed with the linear model presented in [4]. We also give obtained

[1] All these instances are reachable on the site: http://www.bwl.tu-darmstadt.de/bwl3/forsch/projekte/alb/index.htm

results with the well-known heuristic COMSOAL ([2]). Results represent the number of used workstations.

We give the number of obtained optima and the average distance from the optima (defined by formula (4)). The used notations are:

- H is the fitness function value,
- H^* is the optimum,
- *nbinst* is the number of instances.

$$\text{Average distance from optima} = \frac{H - H^*}{nbinst} \tag{4}$$

Table 9-3. Obtained results on literature instances

	Number of obtained optima	Average distance from optima
GA	119	1.32
GA ↔ FRO	144	0.71
GA ↔ (MLB1+MLB2)	150	1.00
GA ↔ (MLB2+FRO)	147	0.81
COMSOAL	65	1.40

We have also tested the hybrid methods on generated data. Their characteristics are presented in Table 9-4 and results in Table 9-5. The effectiveness of the genetic algorithm is limited to small instances. For large instances, hybrid methods are required; combinations with *FRO* provide the best results. We have also applied combinations with stochastic descents. The inconvenient of these combinations is the required time for execution.

In the same way, we have applied this method on real data. Table 9-6 shows characteristics of two instances and Table 9-7 gives results. In these cases, the goal is to modify an initial balancing corresponding to a given period (a given sequence of vehicles) and adapt it to the sequence of the next period ([3]). So an operation is moved if the workstation initially carrying it is different of the one performing it in the final solution. On these real instances, all methods seem to be similar.

Table 9-4. Description of generated data

	Inst20	Inst40	Inst100	Inst500
Number of vehicles	50	100	100	250
Number of workstations	6	10	41	100
Number of sections	4	4	9	24
Number of operations	20	40	100	500
Number of ranges of vehicles	5	10	10	25

Table 9-5. Obtained results on generated data

	Inst20	Inst40	Inst100	Inst500
Optimum	*5*	*7*	*18*	*not found*
GA	5	7	29	80
FRO	5	9	20	38
GA ↔ FRO	5	8	19	37
GA ↔ (MLB1+MLB2)	5	7	18	37
GA ↔ (MLB2+FRO)	5	8	18	36
COMSOAL	5	7	18	37

Table 9-6. Real instance characteristics

	Inst 1	Inst 2
Number of vehicles	17021	17625
Number of workstations	19	37
Number of sections	13	36
Number of operations	310	1166
Number of ranges of vehicles	935	1100
Cycle time	1.23	1.37
Operator time	27018.6	26901
Number of violations		
Cycle time (nb of over-loaded work., nb of concerned ranges in average)	878 (4, 219)	2603 (20, 130)
Operator time	0	0
Section length	0	0

Table 9-7. Obtained results on real data

	Inst1	Inst2
GA	20	33
FRO	20	32
GA ↔ FRO	20	32
GA ↔ (MLB1+MLB2)	20	32
GA ↔ (MLB2+FRO)	20	32
COMSOAL	18	32

7. CONCLUSION

We have proposed to combine a genetic algorithm with different heuristics that we have previously presented. Results show the effectiveness of these combinations. We have tested these hybrid methods on several kinds of instances and good results have been obtained.

Our further works consist in solving our problem with a grouping genetic algorithm. And we are interesting to define the relation between the Resource Constrained Project Scheduling Problem and our line balancing problem.

REFERENCES

1. Anderson E. J. and Ferris M. C., 1994. Genetic Algorithms for Combinatorial Optimization: The Assembly Line Balancing Problem. *ORSA Journal on Computing,* 6, 161-174.
2. Arcus A.L., 1966. COMSOAL a computer method of sequencing operations for assembly lines. *International Journal of Production Research,* 4 (4), 259-277.
3. Boutevin C., Gourgand M. and Norre S., 2002. Stochastic algorithms for the line balancing problem in automotive industry. *IFAC bo'02,* July 21-26 2002, Barcelona (Spain).
4. Boutevin C., Gourgand M. and Norre S., 2002. Optimization methods for the line balancing problem. Submitted to the *European Journal of Operational Research.*
5. Corcoran A. L. and Wainwright R. L., 1995. Using LibGA to Develop Genetic Algorithms for Solving Combinatorial Optimization Problems. *The Application Handbook of Genetic Algorithms,* 1, 144-172.
6. Falkenauer E. and Delchambre A., 1992. A Genetic Algorithm for Bin Packing and Line Balancing. *Proceedings of IEEE International Conference on Robotics and Automation (ICRA92),* Los Alamitos, Californa (USA), 1186-1192.
7. Holland J., 1975. *Adaptation in Natural and Artificial Systems.* University of Michigan Press.
8. Kim K.Y., 1996. Sequencing in Mixed Model Assembly Lines: a Genetic Algorithm Approach. *Computers and Operations Research,* 24 (12), 1141-1145.
9. Kim Y.J. and Kim Y.K. and Cho Y., 1998. A Heuristic-Based Genetic Algorithm for Workload Smoothing in Assembly Lines. *Computers Ops Res.,* 25 (2), 99-111.
10. Rekiek B., 2000. Assembly Line Design: *Multiple Objective Grouping Genetic Algorithm and the Balancing of Mixed-model Hybrid Assembly Line.* Ph.D. Thesis, Université Libre de Bruxelles.
11. Rubinovitz J. and Levitin G., 1995. Genetic Algorithm for Assembly Line Balancing. *International Journal of Production Economics,* 41, 444-454.
12. Scholl A., 1999. *Balancing and Sequencing of Assembly Lines.* Physica-Verlag Heidelberg, New York.

This page intentionally left blank

Chapter 10

STABILITY OF OPTIMAL LINE BALANCE WITH GIVEN STATION SET

Yuri N. Sotskov, Alexandre Dolgui, Nadezhda Sotskova, Frank Werner

Abstract: We consider the simple assembly line balancing problem. For an optimal line balance, we investigate its stability with respect to simultaneous independent variations of the processing times of the manual operations. In particular, we prove necessary and sufficient conditions when optimality of a line balance is stable with respect to sufficiently small variations of operation times. We show how to calculate lower and upper bounds for the stability radius, i.e., the maximal value of simultaneous independent variations of operation times with definitely keeping the optimality of line balance.

Key words: assembly line balance, stability analysis.

1. INTRODUCTION

We consider a single-model paced assembly line, which continuously manufactures a homogeneous product in large quantities as in mass production (see [4] for definitions). An assembly line is a sequence of m linearly ordered stations, which are linked by a conveyor belt. A station has to perform the same set of operations repeatedly during the whole life cycle of the assembly line. The set of operations V, which have to be processed by all m stations within one cycle-time c, is fixed. Each operation $i \in V$ is considered as indivisible. All the m stations start simultaneously with the processing of the sequence of their operations and buffers between stations are absent. Technological factors define a partial order on the set of operations, namely, the digraph $G = (V, A)$ with vertices V and arcs A defines a partially ordered set of operations $V = \{1, 2, \ldots, n\}$.

We assume that set V includes operations of two types. More precisely, the non-empty set $\widetilde{V} \subseteq V$ includes all *manual* operations, and the set $V \setminus \widetilde{V}$ includes all *automated* operations. Without loss of generality, we assume that $\widetilde{V} = \{1, 2, \ldots, \widetilde{n}\}$ and $V \setminus \widetilde{V} = \{\widetilde{n}+1, \widetilde{n}+2, \ldots, n\}$ where $1 \le \widetilde{n} \le n$. We use the following notations for the vectors of the operation times: $\widetilde{t} = (t_1, t_2, \ldots, t_{\widetilde{n}})$, $\bar{t} = (t_{\widetilde{n}+1}, t_{\widetilde{n}+2}, \ldots, t_n)$ and $t = (\widetilde{t}, \bar{t}) = (t_1, t_2, \ldots, t_n)$.

The *Simple Assembly Line Balancing Problem* is to find an optimal balance of the assembly line for a given number m of stations, i.e., to find a feasible assignment of all operations V to exactly m stations in such a way that the cycle-time c is minimal. In [1, 4], the abbreviation SALBP-2 is used for this problem.

Let the set of operations $V_k^{b_r}$ be assigned to station S_k, $k \in \{1, 2, \ldots, m\}$. Assignment b_r: $V = V_1^{b_r} \cup V_2^{b_r} \cup \ldots \cup V_m^{b_r}$ of operations V to m ordered stations $S_1, S_2, \ldots, S_m$ (where $V_u^{b_r} \cap V_w^{b_r} = \varnothing$, $1 \le u < w \le m$) is called a *line balance,* if the following two conditions hold.

1. *Assignment b_r does not violate the partial order given by digraph $G=(V, A)$, i.e., inclusion $(i, j) \in A$ implies that operation i is assigned to station S_k and operation j is assigned to station S_l such that* $1 \le k \le l \le m$.

2. *Assignment b_r uses exactly m stations: $V_k^{b_r} \neq \varnothing$, $k \in \{1, 2, \ldots, m\}$.*

Line balance b_r is *optimal* if along with conditions 1 and 2, it has the minimal cycle-time. We denote the cycle-time for line balance b_r with the vector t of operation times as $c(b_r, t)$:

$$c(b_r, t) = max_{k=1}^{m} \sum_{i \in V_k^{b_r}} t_i \,.$$

Optimality of line balance b_0 with vector t of operation times may be defined as the following condition 3.

3. $c(b_0, t) = min\{c(b_r, t) : b_r \in B\}$, *where $B = \{b_0, b_1, \ldots, b_h\}$ is the set of all line balances.*

If $j \in \widetilde{V}$, then the processing time t_j of operation j is a given non-negative real number: $t_j \ge 0$. However, the value of the manual operation time t_j can vary during the life cycle of the assembly line and can even be equal to zero. A zero operation time t'_j means that operation $j \in V_k^{b_r} \cap \widetilde{V}$ is processed by an *additional worker* simultaneously (in parallel) with other operations assigned to station S_k in such a way that the processing of operation j does not increase the station time for S_k:

$$\sum_{i \in V_k^{b_r}} t'_i = \sum_{i \in V_k^{b_r} \setminus \{j\}} t'_i$$

Obviously, the latter equality is only possible if $t'_j = 0$.

If $i \in V\backslash\widetilde{V}$, then operation time t_i is a real number fixed during the whole life cycle of the assembly line. We can assume that $t_i > 0$ for each automated operation $i \in V\backslash\widetilde{V}$. Indeed, an operation with *fixed zero* processing time (if any) has no influence on the solution of SALBP-2, and therefore in what follows, we will consider automated operations, which have only strictly positive processing times.

In contrast to usual *stochastic* problems (see survey [2]), we do not assume any probability distribution known in advance for the random processing times of the manual operations. Moreover, this chapter does not deal with concrete algorithms for constructing an optimal line balance in a stochastic environment. It is assumed that the optimal line balance b_0 is already constructed for the given vector $t = (t_1, t_2, ..., t_n)$ of the operation times. Our aim is to investigate the stability of the optimality of a line balance b_0 with respect to independent variations of the processing times of all manual operations $\widetilde{V} = \{1, 2, \ldots, \widetilde{n}\}$ or a portion of the manual operations. More precisely, we investigate the *stability radius* of an optimal line balance b_0, which may be interpreted as the maximum of simultaneous independent variations of the manual operation times with definitely keeping optimality of line balance b_0.

It will be assumed that all operation times t_i, $i \in V$, are real numbers in contrast to the usual assumption that they are integral numbers (see [4]). We need this assumption for the sake of appropriate definitions introduced in Section 2 for a sensitivity analysis. In Section 3, we prove necessary and sufficient conditions for the existence of an *unstable* optimal line balance, i.e., when its stability radius is equal to zero. In Section 4, we show how to calculate the exact value of the stability radius or its upper bound. An algorithm for selecting all stable optimal line balances is discussed in Section 4. Concluding remarks are given in Section 5.

2. DEFINITION OF THE STABILITY RADIUS

The main question under consideration may be formulated as follows. How much can all components of vector $\widetilde{t}$ simultaneously and independently be modified such that the given line balance b_0 remains definitely optimal? To answer this question, we study the notion of the stability radius. The stability radius of an optimal line balance may be defined similarly to the stability radius of an optimal schedule introduced in [5] for a machine scheduling problem. (A survey of known results on sensitivity analysis in machine scheduling is presented in [8].) On the one hand, if the stability radius of line balance b_0 is strictly positive, then any simultaneous independent changes of the operation times t_j, $j\in\widetilde{V}$, within the ball with this radius definitely keep optimality of line balance b_0. On the other hand, if the stability radius of line balance b_0 is equal to zero, then even small changes of the processing times of the manual operations may deprive optimality of line balance b_0.

We consider the space $R^{\tilde{n}}$ of real vectors $\tilde{t} = (t_1, t_2, ..., t_{\tilde{n}})$ with the Chebyshev metric. So, the distance $d(\tilde{t},\tilde{t}^*)$ between vector $\tilde{t}\in R^{\tilde{n}}$ and vector $\tilde{t}^* = (t_1^*, t_2^*, \ldots, t_{\tilde{n}}^*) \in R^{\tilde{n}}$ is calculated as follows:

$$d(\tilde{t},\tilde{t}^*) = \max\{\,|t_i - t_i^*| : i\in\widetilde{V}\},$$

where $\left|t_i - t_i^*\right|$ denotes the absolute value of the difference $t_i - t_i^*$. We also consider the space of non-negative real vectors:

$$R_+^{\tilde{n}} = \{\tilde{t}\in R^{\tilde{n}}: t_i \geq 0,\ i\in\widetilde{V}\}.$$

Let $B(t)$ denote the set of all line balances in the set B, which are optimal for the given vector t of the operation times. The formal definition of the stability radius of an optimal line balance may be given as follows.

Definition 1: *The closed ball $O_\rho(\tilde{t})$ in the space $R^{\tilde{n}}$ with the radius $\rho\in R_+^1$ and the center $\tilde{t}\in R_+^{\tilde{n}}$ is called a stability ball of the line balance $b_0\in B(t)$, if for each vector $t^* = (\tilde{t}^*, \bar{t})$ of the operation times with $\tilde{t}^*\in O_\rho(\tilde{t})\cap R_+^{\tilde{n}}$ line balance b_0 remains optimal. The maximal value of the radius ρ of a stability ball $O_\rho(\tilde{t})$ of the line balance b_0 is called the stability radius denoted by $\rho_{b_0}(t)$.*

In Definition 1, vector $\bar{t} = (t_{\tilde{n}+1}, t_{\tilde{n}+2}, \ldots, t_n)$ of the automated operation times and vector $t = (\tilde{t},\bar{t}) = (t_1, t_2, \ldots, t_n)$ of all operation times are fixed, while vector $\tilde{t}^* = (t_1^*, t_2^*, ..., t_{\tilde{n}}^*)$ of the manual operation times may vary within the intersection of the closed ball $O_\rho(\tilde{t})$ with the space $R_+^{\tilde{n}}$. To illustrate the above notations, we use the following example of SALBP-2.

Let $m = 3$, $\tilde{n} = 2$, $n = 7$ and $t = (\tilde{t},\bar{t}) = (3, 1, 6, 3, 7, 2, 4)$. Thus, set $\widetilde{V} = \{1, 2\}$ is the set of manual operations, and set $V\backslash\widetilde{V} = \{3, 4, 5, 6, 7\}$ is the set of automated operations. The digraph $G = (V, A)$ and the operation times are represented in Figure 10-1.

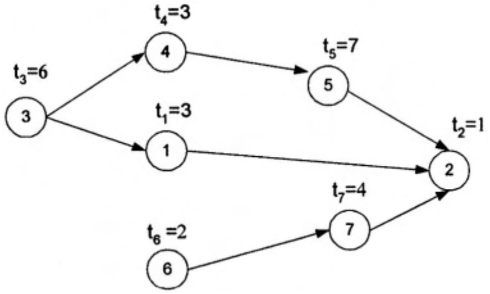


Figure 10-1. Digraph *G = (V, A)* and operation times

Next, we show that the following line balance b_0:

$V_1^{b_0} = \{3, 4\},\ V_2^{b_0} = \{1, 6, 7\},\ V_3^{b_0} = \{2, 5\}$

is optimal. To this end, we can use the obvious lower bound (1) for the minimal cycle-time.

If all operation times t_i, $i \in V$, are integral numbers, then

$$min\{c(b_r, t) : b_r \in B\} \geq \left\lceil \sum_{i=1}^{n} t_i \middle/ m \right\rceil . \tag{1}$$

Hereafter, $\lceil a \rceil$ denotes the smallest integral number greater than or equal to a. For the above line balance b_0, we have the following equalities:

$$\left\lceil \sum_{i=1}^{n} t_i \middle/ m \right\rceil = \lceil 26/3 \rceil = 9 = c(b_0, t),$$

which imply that b_0 is an optimal line balance since $c(b_0, t)$ is equal to the right-hand side of inequality (1).

Let $\tilde{V}_k^{b_r}$ denote the subset of all manual operations of set $V_k^{b_r}$. For each optimal line balance $b_r \in B(t)$, we can define a set $W(b_r)$ of all subsets $\tilde{V}_k^{b_r}$, $k \in \{1, 2, ..., m\}$, such that

$$\sum_{i \in V_k^{b_r}} t_i = c(b_0, t).$$

It should be noted that set $W(b_r)$ may include the empty set as its element. E.g., in the example presented in Figure 10-1 for the optimal line balance $b_0 \in B(t)$, we have $W(b_0) = \{\emptyset, \{1\}\}$ since $\tilde{V}_1^{b_0} = \emptyset$, $\tilde{V}_2^{b_0} = \{1\}$, and

$$\sum_{i \in V_1^{b_0}} t_i = \sum_{i \in V_2^{b_0}} t_i = c(b_0, t) = 9.$$

Note that the empty set may be considered as a proper subset of any non-empty set, e.g., we can write $\widetilde{V}_1^{b_0} \subset \widetilde{V}_2^{b_0}$.

3. ZERO STABILITY RADIUS

In this section, we derive necessary and sufficient conditions for the existence of an unstable optimal line balance $b_0 \in B(t)$.

Theorem 1: *Let inequality $t_i > 0$ hold for each manual operation $i \in \widetilde{V}$. Then for line balance $b_0 \in B(t)$, equality $\rho_{b_0}(t) = 0$ holds if and only if there exists a line balance $b_r \in B(t)$ such that condition $W(b_0) \subseteq W(b_r)$ does not hold.*

Proof: *Sufficiency.* Let there exist a line balance $b_r \in B(t)$, for which condition $W(b_0) \subseteq W(b_r)$ does not hold.

Hence, there exists at least one set $\widetilde{V}_k^{b_0} \in W(b_0)$, which does not belong to the set $W(b_r)$. We have to consider the following three possible cases (i), (ii) and (iii).

Case (i): There exists a set $\widetilde{V}_l^{b_r} \in W(b_r)$ such that $\widetilde{V}_k^{b_0}$ is a proper subset of set $\widetilde{V}_l^{b_r}$, i.e. $\widetilde{V}_k^{b_0} \subset \widetilde{V}_l^{b_r}$ and inequality

$$|\widetilde{V}_k^{b_0}| > |\widetilde{V}_l^{b_r} \setminus \widetilde{V}_k^{b_0}| \qquad (2)$$

holds.

From the inclusion $\widetilde{V}_k^{b_0} \subset \widetilde{V}_l^{b_r}$ it follows that set $\widetilde{V}_l^{b_r} \setminus \widetilde{V}_k^{b_0}$ is a non-empty set. Let ε be any arbitrarily small real number such that $\varepsilon > 0$ and $\varepsilon \leq t_i$, $i \in \widetilde{V}$. We can construct the following vector $\widetilde{t}^{\varepsilon} = (t_1^{\varepsilon}, t_2^{\varepsilon}, \ldots, t_{\widetilde{n}}^{\varepsilon})$, where

$$t_i^{\varepsilon} = \begin{cases} t_i + \varepsilon, & \text{if} \quad i \in \widetilde{V}_k^{b_0}, \\ t_i - \varepsilon, & \text{if} \quad i \in \widetilde{V}_l^{b_r} \setminus \widetilde{V}_k^{b_0}. \end{cases}$$

For all other manual operations $p \in \widetilde{V} \setminus \widetilde{V}_l^{b_r}$, we set $t_p^{\varepsilon} = t_p$. Note that due to assumption $t_i > 0$ for each operation $i \in \widetilde{V}$, all components of vector $\widetilde{t}^{\varepsilon}$ are non-negative, and therefore $\widetilde{t}^{\varepsilon} \in R_+^{\widetilde{n}}$. Inequality (2) implies $\widetilde{V}_k^{b_0} \neq \varnothing$. Therefore, since $t_p^{\varepsilon} \leq t_p$ for each operation $p \in \widetilde{V} \setminus \widetilde{V}_k^{b_0}$, we obtain

$$c(b_0, t^{\varepsilon}) = \sum_{i \in V_k^{b_0}} t_i^{\varepsilon} = c(b_0, t) + \varepsilon \, | \widetilde{V}_k^{b_0} | \tag{3}$$

where $t^{\varepsilon} = (\widetilde{t}^{\,\varepsilon}, \bar{t}\,)$. Due to inequality (2) and equalities $t_p^{\varepsilon} = t_p$, $p \in \widetilde{V} \setminus \widetilde{V}_l^{b_r}$, we obtain

$$c(b_r, t^{\varepsilon}) = \sum_{i \in V_l^{b_r}} t_i^{\varepsilon} = c(b_r, t) + \varepsilon \, | \widetilde{V}_k^{b_0} | - \varepsilon \, | \widetilde{V}_l^{b_r} \setminus \widetilde{V}_k^{b_0} |. \tag{4}$$

Since set $\widetilde{V}_l^{b_r} \setminus \widetilde{V}_k^{b_0}$ is a non-empty set and $c(b_0, t) = c(b_r, t)$, equalities (3) and (4) imply the strict inequality $c(b_0, t^{\varepsilon}) > c(b_r, t^{\varepsilon})$.

As a result, we conclude that for any arbitrarily small $\varepsilon > 0$ ($\varepsilon \leq t_i$, $i \in \widetilde{V}$), there exists a vector $\widetilde{t}^{\,\varepsilon} \in R_+^{\widetilde{n}}$ such that $d(\widetilde{t}\,, \widetilde{t}^{\,\varepsilon}) = \varepsilon$ and $c(b_0, t^{\varepsilon}) > c(b_r, t^{\varepsilon})$. Therefore, we obtain $b_0 \notin B(t^{\varepsilon})$. Since vector $\widetilde{t}^{\,\varepsilon}$ may be as close to vector $\widetilde{t}$ as desired, we obtain equality $\rho_{b_0}(t) = 0$ in case (i).

<u>Case (ii):</u> There exists a set $\widetilde{V}_l^{b_r} \in W(b_r)$ such that $\widetilde{V}_k^{b_0}$ is a proper subset of set $\widetilde{V}_l^{b_r}$ and inequality $| \widetilde{V}_k^{b_0} | \leq | \widetilde{V}_l^{b_r} \setminus \widetilde{V}_k^{b_0} |$ holds.

Since $\widetilde{V}_k^{b_0} \subset \widetilde{V}_l^{b_r}$, there exists at least one operation $j \in \widetilde{V}_l^{b_r}$, which does not belong to set $\widetilde{V}_k^{b_0} : j \notin \widetilde{V}_k^{b_0}$. For any arbitrarily small $\varepsilon > 0$ ($\varepsilon \leq t_i$, $i \in \widetilde{V}$), we can construct vector $\widetilde{t}^{\,(\varepsilon)} = (t_1^{(\varepsilon)}, t_2^{(\varepsilon)}, \ldots, t_{\widetilde{n}}^{(\varepsilon)})$, where

$$t_i^{(\varepsilon)} = \begin{cases} t_i + \varepsilon, \ \textit{if} \quad i \in \widetilde{V}_k^{b_0}, \\ t_i - \varepsilon, \ \textit{if} \quad i \in \{j\} \cup \{\widetilde{V} \setminus \widetilde{V}_l^{b_r}\}, \\ t_i, \ \textit{if} \quad i \in \widetilde{V}_l^{b_r} \setminus \{\widetilde{V}_k^{b_0} \cup \{j\}\}. \end{cases}$$

Since $t_p^{(\varepsilon)} \leq t_p$ for each operation $p \in \widetilde{V} \setminus \widetilde{V}_k^{b_0}$, the following equalities must hold:

$$c(b_0, t^{(\varepsilon)}) = \sum_{i \in V_k^{b_0}} t_i^{(\varepsilon)} = c(b_0, t) + \varepsilon \, | \widetilde{V}_k^{b_0} |, \tag{5}$$

where $t^{(\varepsilon)} = (\widetilde{t}^{\,(\varepsilon)}, \bar{t}\,)$.

Next, we consider two possible subcases: either $\widetilde{V}_k^{b_0} \neq \varnothing$ or $\widetilde{V}_k^{b_0} = \varnothing$.

If $\widetilde{V}_k^{b_0} \neq \varnothing$, then due to equalities (5) and $c(b_0, t) = c(b_r, t)$, we obtain

$$c(b_r, t^{(\varepsilon)}) = \sum_{i \in V_l^{b_r}} t_i^{(\varepsilon)} = c(b_r, t) + \varepsilon \,|\, \widetilde{V}_k^{b_0} \,| - \varepsilon < c(b_0, t^{(\varepsilon)}).$$

If $\widetilde{V}_k^{b_0} = \varnothing$, then we can conclude that set $W(b_r)$ does not contain the empty set as its element. Indeed, if $\widetilde{V}_q^{b_r} = \varnothing$ for some index $q \neq l$, then $\varnothing = \widetilde{V}_k^{b_0} \subseteq \widetilde{V}_q^{b_r}$, which contradicts to the above assumption about the set $\widetilde{V}_k^{b_0}$. Therefore, due to equalities $t_p^{(\varepsilon)} = t_p - \varepsilon$, $p \in \widetilde{V} \setminus \widetilde{V}_l^{b_r}$, we obtain

$$c(b_r, t^{(\varepsilon)}) = \sum_{i \in V_l^{b_r}} t_i^{(\varepsilon)} = \mathrm{c(b_r, t) - \varepsilon}. \tag{6}$$

In the case of an empty set $\widetilde{V}_k^{b_0}$, condition (5) turns into equality

$$c(b_0, t^{(\varepsilon)}) = c(b_0, t). \tag{7}$$

From (6) and (7), it follows that $c(b_r, t^{(\varepsilon)}) < c(b_0, t^{(\varepsilon)})$. Thus, using the same arguments for vector $\widetilde{t}^{(\varepsilon)}$ as for vector $\widetilde{t}^{\,\varepsilon}$ (see case (i)), we conclude that $b_0 \notin B(t^{(\varepsilon)})$, and therefore, $\rho_{b_0}(t) = 0$ in case (ii) as well.

<u>Case (iii):</u> There is no set $\widetilde{V}_l^{b_r} \in W(b_r)$ such that $\widetilde{V}_k^{b_0}$ is a subset of set $\widetilde{V}_l^{b_r}$.

It is clear that $\widetilde{V}_k^{b_0} \neq \varnothing$ (otherwise $\widetilde{V}_k^{b_0} = \varnothing \subseteq \widetilde{V}_l^{b_r} \in W(b_r)$). For any arbitrarily small real $\varepsilon > 0$ ($\varepsilon \leq t_i$, $i \in \widetilde{V}$), we can construct the following vector $\widetilde{t}^{[\varepsilon]} = (t_1^{[\varepsilon]}, t_2^{[\varepsilon]}, \ldots, t_{\widetilde{n}}^{[\varepsilon]}) \in R_+^{\widetilde{n}}$, where

$$t_i^{[\varepsilon]} = \begin{cases} t_i + \varepsilon, & \text{if} \quad i \in \widetilde{V}_k^{b_0}, \\ t_i, & \text{if} \quad i \in \widetilde{V} \setminus \widetilde{V}_k^{b_0}. \end{cases}$$

It is easy to convince that

$$c(b_0, t^{[\varepsilon]}) = \sum_{i \in V_k^{b_0}} t_i^{[\varepsilon]} = c(b_0, t) + \varepsilon \,|\, \widetilde{V}_k^{b_0} \,| > c(b_r, t^{[\varepsilon]}).$$

The latter inequality follows from the fact that set $\widetilde{V}_k^{b_0}$ is not contained in any set $\widetilde{V}_l^{b_r} \in W(b_r)$ and $\widetilde{t}_i^{[\varepsilon]} = t_i$ for each operation $i \in \widetilde{V} \setminus \widetilde{V}_k^{b_0}$. Using similar arguments for vector $\tilde{t}^{[\varepsilon]}$ as for vector $\tilde{t}^{\varepsilon}$ (see case (i)), we conclude that $b_0 \notin B(t^{[\varepsilon]})$, and therefore, $\rho_{b_0}(t) = 0$ in case (iii) as well.

Necessity: Assume that there does not exist a line balance $b_r \in B(t)$, for which condition $W(b_0) \subseteq W(b_r)$ does not hold.

In other words, either $B(t) = \{b_0\}$ or for any line balance $b_r \in B(t) \setminus \{b_0\}$, condition $W(b_0) \subseteq W(b_r)$ holds. Thus, we have to consider the following two possible cases (j) and (jj).

Case (j): $B(t) = \{b_0\}$.

Let us compare line balance b_0 with an arbitrary line balance $b_s \in B \setminus B(t)$. Since line balance b_s is not optimal for vector t of the operation times, the strict inequality $c(b_s, t) > c(b_0, t)$ must hold. Therefore, for any vector $\tilde{t}^{\delta} \in R_+^{\tilde{n}}$ with $d(\tilde{t}, \tilde{t}^{\delta}) = \delta > 0$, the opposite inequality $c(b_s, t^{\delta}) < c(b_0, t^{\delta})$ may hold for vector $t^{\delta} = (\tilde{t}^{\delta}, \bar{t})$ only if

$$\delta > \Delta(b_s) = \frac{c(b_s,t) - c(b_0,t)}{\tilde{n}}. \tag{8}$$

Indeed, one can overcome the strictly positive difference $c(b_s, t) - c(b_0, t)$ only via changing the processing times t_i of the $\tilde{n}$ manual operations $i \in \tilde{V}$. Recall that $\tilde{n} \geq 1$. Due to bound (8), the desired vector $\tilde{t}^{\delta} \in R_+^{\tilde{n}}$ cannot be arbitrarily close to vector $\tilde{t}$.

Since bound (8) must hold for any non-optimal line balance, we conclude that condition (9) holds for the desired vector $\tilde{t}^{\delta} \in R_+^{\tilde{n}}$:

$$d(\tilde{t}, \tilde{t}^{\delta}) > \Delta = \min\{\Delta(b_s) : b_s \in B \setminus B(t)\} > 0. \tag{9}$$

As a result we obtain $\rho_{b_0}(t) \geq \Delta > 0$.

Case (jj): $B(t) \setminus \{b_0\} \neq \varnothing$.

Obviously, the lower bound (9) for the distance between vector $\tilde{t}$ and the desired vector is correct in case (jj) as well. Therefore, we have to compare line balance b_0 only with other optimal line balances.

Let b_r be an arbitrary line balance from the set $B(t) \setminus \{b_0\}$. Due to condition $W(b_0) \subseteq W(b_r)$, there exists a subset $W^*(b_r)$ of the set $W(b_r)$ such that $W(b_0) = W^*(b_r)$.

If there exists an index $k \in \{1, 2, \ldots, m\}$ such that

$$\sum_{i \in V_k^{b_0}} t_i < c(b_0,t), \tag{10}$$

then we set

$$\delta(b_0) = \{c(b_0,t) - \max\{\sum_{i\in V_k^{b_0}} t_i : \tilde{V}_k^{b_0} \notin W(b_0)\}\}/\tilde{n}\,\cdot$$

Due to (10), the strict inequality $\delta(b_0) > 0$ holds.

If $\sum_{i\in V_k^{b_0}} t_i = c(b_0,t)$ for each index $k \in \{1, 2, \ldots, m\}$, then we set $\delta(b_0) = min\{t_i : i \in \tilde{V}\}$.

We consider an arbitrarily small real number δ, where $0 < \delta \leq \delta(b_0)$ and an arbitrary vector $\tilde{t}^{\delta} \in R_+^{\tilde{n}}$, for which equality $d(\tilde{t},\tilde{t}^{\delta}) = \delta$ holds. Hereafter, we use the notation

$$t^{\delta}(V') = \sum_{i\in V'} t_i^{\delta}\,, \tag{11}$$

where t^{δ} denotes a vector the components of which are used in the right-hand side of equality (11). Inequality $\delta \leq \delta(b_0)$ implies

$$c(b_0, t^{\delta}) = \max\{t^{\delta}(V_k^{b_0}) : \tilde{V}_k^{b_0} \in W(b_0)\}, \tag{12}$$

where $t^{\delta} = (\tilde{t}^{\delta}, \bar{t})$. For any line balance $b_r \in B(t) \setminus \{b_0\}$, we obtain

$$c(b_r, t^{\delta}) = \max\{\max\{t^{\delta}(V_l^{b_r}): \tilde{V}_l^{b_r} \in W^*(b_r) = W(b_0)\},$$
$$\max\{t^{\delta}(V_q^{b_r}): \tilde{V}_q^{b_r} \in W(b_r)\setminus W^*(b_r)\}\}. \tag{13}$$

From (12) and (13), it follows that $c(b_0, t^{\delta}) \leq c(b_r, t^{\delta})$. As a consequence, for any $\delta > 0$ ($\delta \leq \delta(b_0)$), inequality $c(b_0, t^{\delta}) \leq c(b_r, t^{\delta})$ holds for an arbitrary vector $t^{\delta} = (\tilde{t}^{\delta}, \bar{t})$ with distance $d(\tilde{t},\tilde{t}^{\delta}) = \delta > 0$.

From the latter statement and inequalities (9), it follows that $\rho_{b_0}(t) \geq min\{\Delta, \delta(b_0)\} > 0$. Thus, Theorem 1 is proven.

□

The above proof implies the following corollaries.

Corollary 1: If $B(t) = \{b_0\}$, then $\rho_{b_0}(t) > 0$.

Corollary 2: If $\rho_{b_0}(t) > 0$, then $\rho_{b_0}(t) \geq min\{\Delta, \delta(b_0)\}$.

The latter claim gives a lower bound for a strictly positive stability radius.

4. STABLE OPTIMAL LINE BALANCE

Next, we present an algorithm for selecting the set of stable optimal line balances $B^*(t) \subseteq B(t)$, i.e., all optimal line balances $b_r \in B^*(t)$ with strictly positive stability radii: $\rho_{b_r}(t) > 0$.

Algorithm 1

```
INPUT: $G=(V, A)$, $t = (\tilde{t}, \bar{t})$.
OUTPUT: Set $B^*(t)$ of all stable optimal line balances.
1.Construct the optimal line balances $B(t) = \{b_0, ..., b_{h^*}\}$,
      $0 \leq h^* \leq h$.
2.Construct set $W(b_r)$ for each optimal line balance $b_r \in B(t)$.
      Set $B^*(t) = \emptyset$.
3.DO for r = 0, h*
      DO for $s = 0, h^*; b_s \neq b_r$
         IF condition $W(b_r) \subseteq W(b_s)$ does not hold, THEN GOTO 5.
         IF $b_s = b_{h^*}$, THEN GOTO 4.
      END
4. Line balance $b_r$ is stable: $\rho_{b_r}(t) > 0$.
      Set $B^*(t) := B^*(t) \cup \{b_r\}$. GOTO 6.
5. Line balance $b_r$ is unstable: $\rho_{b_r}(t) = 0$.
6. END
```

Due to Theorem 1, all stable optimal line balances are selected by Algorithm 1. Within step 1 of Algorithm 1, one has to solve problem SALBP-2 which is binary NP-hard even if $m = 2$ and $A = \emptyset$. The latter claim may be easily proven by a polynomial reduction of the NP-complete *partition* problem to SALBP-2 with $m = 2$ (see e.g. [4]). To reduce the calculations in steps 2 – 6, we can consider a proper subset of set $B(t)$ instead of the whole set.

Returning to the example of problem SALBP-2 presented in Section 2 (see Figure 10-1), we can construct the set $B^*(t) = \{b_0, b_1, b_2\}$ of all optimal line balances, where

$$V_1^{b_1} = \{1, 3\},\ V_2^{b_1} = \{4, 6, 7\},\ V_3^{b_1} = \{2, 5\};$$

$$V_1^{b_2} = \{3, 4\},\ V_2^{b_2} = \{5, 6\},\ V_3^{b_2} = \{1, 2, 7\}.$$

We find the sets $W(b_1) = \{\emptyset, \{1\}\}$ and $W(b_2) = \{\emptyset\}$ since $\widetilde{V}_1^{b_1} = \{1\}$, $\widetilde{V}_2^{b_1} = \emptyset$ and $\widetilde{V}_1^{b_2} = \widetilde{V}_2^{b_2} = \emptyset$. Due to Theorem 1, we obtain equality $\rho_{b_0}(t) = 0$ since condition $W(b_0) \subseteq W(b_2)$ does not hold for line balance $b_2 \in B(t)$. Similarly, due to Theorem 1, $\rho_{b_1}(t) = 0$ since condition $W(b_1) \subseteq W(b_2)$ does not hold. The only optimal line balance with a strictly positive stability radius is line balance b_2. Indeed, for any optimal line balance $b_r \in B(t)$, condition $W(b_2) \subseteq W(b_r)$ holds. Thus, Algorithm 1 gives the singleton $B^*(t) = \{b_2\}$.

Next, we show how to use Theorem 1 for the calculation of the exact value of a strictly positive stability radius $\rho_{b_s}(t)$ for line balance $b_s \in B^*(t)$. For calculating $\rho_{b_s}(t)$, we have to find a line balance $b_r \in B$ and a vector $\widetilde{t}' = (t'_1, t'_2, \ldots, t'_{\widetilde{n}}) \in R_+^{\widetilde{n}}$ such that

$$c(b_r, t') < c(b_s, t'), \tag{14}$$

where $t' = (\widetilde{t}', \bar{t})$ and vector $\widetilde{t}'$ is the closest vector to vector $\widetilde{t}$, for which inequality (14) holds.

Since value $c(b_r, t)$ linearly depends on the components of vector $\widetilde{t}$, before reaching inequality (14) via a continuous change of the components of vector $\widetilde{t}$, we first reach equality $c(b_r, t^r) = c(b_s, t^r)$ for some new vector $t^r = (\widetilde{t}^r, \bar{t})$, for which the optimal line balance b_s becomes not stable, i.e., equality (15) holds:

$$\rho_{b_s}(t^r) = 0. \tag{15}$$

Let $W(b_r, t^r)$ denote the set of all subsets $\widetilde{V}_k^{b_r}$, $k \in \{1, 2, \ldots, m\}$, with the valid equality

$$t^r(V_k^{br}) = c(b_0, t^r). \tag{16}$$

Due to equality (15) (see Theorem 1), there exists a line balance $b_r \in B(t^r)$ such that condition $W(b_s, t^r) \subseteq W(b_r, t^r)$ does not hold. Therefore, using the same arguments as in the sufficiency proof of Theorem 1 (see cases (i), (ii) and (iii)), we can construct a vector $\widetilde{t}'$ for which inequality (14) holds and $d(\widetilde{t}', \widetilde{t}^r) = \varepsilon > 0$, where ε may be chosen as small as desired.

Thus, the calculation of $\rho_{b_s}(t)$ for line balance $b_s \in B^*(t)$ is reduced to the construction of the closest vector t^r to vector t for which equality (15) holds.

Next, we demonstrate this construction for the example shown in Figure 10-1. Namely, we consider two possibilities (see case (l) and case (ll)) how we can reach equality (15) for line balance $b_s = b_2 \in B(t) = \{b_0, b_1, b_2\}$.

<u>Case (l)</u>: $b_r \in B(t)$

Since $\rho_{b_2}(t') > 0$, condition

$$W(b_2, t) \subseteq W(b_r, t) \tag{17}$$

holds for any optimal line balance $b_r \in B(t)$ (see Theorem 1). In order to get equality (15) we have to violate a condition like (17), namely, for a new vector $t^r = (\tilde{t}^r, \bar{t})$ condition

$$W(b_2, t^r) \subseteq W(b_r, t^r) \tag{18}$$

must be incorrect. To violate condition (18), we can include a new element into set $W(b_2)$ or delete corresponding elements from the set $W(b_r)$. The latter possibility cannot be realized since set $W(b_r)$ includes the empty set as its element. Therefore, we only can include a new element into the set $W(b_2)$.

It is clear that the only candidate for such an inclusion is the subset $\tilde{V}_3^{b2} = \{1, 2\}$ of set $V_3^{b_2} = \{1, 2, 7\}$. If we set $t_2^r = t_2 + 1 = 2$ and $t_1^r = t_1$, then we obtain $W(b_2, t^r) = W(b_2, t) \cup \{\tilde{V}_3^{b2}\}$, $W(b_2, t^r) = \{\emptyset, \{1, 2\}\}$, and condition (18) does not hold. Therefore, $\rho_{b2}(t^r) \geq d(t, t^r) = 1$.

<u>Case (ll)</u>: $b_r \in B \setminus B(t)$

In this case, we have to make line balance $b_r \in B \setminus B(t)$ optimal for a new vector $t^r = (\tilde{t}^r, \bar{t})$ violating condition (18). It is easy to see that line balance $b_3 \in B \setminus B(t)$:

$V_1^{b3} = \{3, 6\}$, $V_2^{b3} = \{1, 4, 7\}$, $V_3^{b3} = \{2, 5\}$

may be included into set $B(t^r)$ with a cycle time equal to 9. To this end, we can set $t_1^r = t_l - 1 = 2$, $t_2^r = t_2$, and obtain $B(t^r) = B(t) \cup \{b_3\}$.

Thus, in both cases (l) and (ll), we have $\rho_{b_2}(t^r) \geq d(t, t^r) = 1$. It is easy to convince that vectors $\tilde{t}^r$ constructed in case (l) and case (ll) are the closest to vector $\tilde{t}$ with equality (15) being correct. Therefore, due to Theorem 1 we obtain $\rho_{b_2}(t) = 1$.

The same cases (1) and (11) have to be considered for calculating the stability radius for any line balance $b_s \in B^*(t)$. In case (1), we have to compare line balance b_s with all line balances $b_r \in B(t)$ and calculate the following upper bound for the stability radius $\rho_{b_s}(t)$:

$$\max\left\{ \frac{c(b_s,t)-t(V_k^{b_s})}{|\tilde{V}_u^{b_s} \oplus \tilde{V}_k^{b_r}|} : |\tilde{V}_u^{b_s} \oplus \tilde{V}_k^{b_r}| \geq 1, \tilde{V}_u^{b_s} \bigcap \tilde{V}_k^{b_r} \neq \emptyset \right\},$$

where the sign $\oplus$ denotes the direct sum of two sets.

In case (11), we have to compare line balance b_s with line balances $b_r \in B \setminus B(t)$ and calculate the following upper bound for the stability radius $\rho_{b_s}(t)$:

$$\max\left\{ \frac{t(V_k^{b_r})-c(b_s,t)}{|\tilde{V}_u^{b_s} \oplus \tilde{V}_k^{b_r}|} : |\tilde{V}_u^{b_s} \oplus \tilde{V}_k^{b_r}| \geq 1, \tilde{V}_u^{b_s} \bigcap \tilde{V}_k^{b_r} \neq \emptyset \right\}.$$

If all competitive line balances will be compared with line balance $b_s \in B^*(t)$, then we calculate the exact value of $\rho_{b_s}(t)$, otherwise we obtain an upper bound for the stability radius. In order to restrict the set of line balances which have to be compared with b_s, we can use an approach similar to the one derived in [7] for the stability radius of an optimal schedule.

5. CONCLUSION

We can give two remarks how to restrict the set of optimal line balances considered in Algorithm 1. First, it should be noted that we do not distinguish line balances which have only different orders of the subsets V_k^b, $k \in \{1, 2, ..., m\}$, but their set of subsets $\{V_1^b, V_2^b, ..., V_m^b\}$ is the same. Second, in practice not all optimal line balances are suitable for a realization since not only precedence constraints defined by the arc set A have to be taken into account. Therefore, the cardinality of set $B(t)$ used in Algorithm 1 may be essentially smaller than $|B(t)|$.

It is easy to show that SALBP-2 may be considered as the problem of scheduling n partially ordered jobs on m parallel (identical) machines with the makespan criterion. In [3], this problem is denoted as $P \mid prec \mid C_{\max}$. Therefore, the above results for an optimal line balance may be interpreted as results on the stability analysis of an optimal schedule for problem $P \mid prec \mid C_{\max}$.

At the stage of the design of the assembly line, another mathematical problem (denoted as SALBP-1) has to be considered. Problem SALBP-1 is to minimize the number of stations when the cycle-time is given and fixed. The stability of feasibility and optimality of a line balance for problem SALBP-1 have been considered in [6].

ACKNOWLEDGEMENTS

This research was supported by INTAS (Project 00-217) and Fundamental Research Found of Belarus (Project Ф03MC-039).

REFERENCES

1. Baybars I., 1986. A survey of exact algorithms for the simple assembly line balancing problem, *Management Science,* 32, 909-932.
2. Erel E., Sarin S.C., 1998. A survey of the assembly line balancing procedures, *Production Planning & Control,* 9, 414 – 434.
3. Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G., Shmoys D.B., 1993. Sequencing and scheduling: algorithms and complexity, in: *Handbook in Operations Research and Management Science 4: Logistic of Production and Inventory,* edited by Graves S.C., Rinnooy Kan A.H.G., and Zipkin P., North-Holland, 445-522.
4. Scholl A. 1999. *Balancing and Sequencing of Assembly Lines,* Heidelberg: Physica-Verlag, A Springer-Verlag Company.
5. Sotskov Yu.N., 1991. Stability of an optimal schedule, *European Journal of Operational Research,* 55, 91-102.
6. Sotskov Yu.N., Dolgui A., 2001. Stability radius of the optimal assembly line balance with fixed cycle time, in: *Proceedings of the IEEE Conference ETFA '2001,* 623-628.
7. Sotskov Yu.N., Sotskova N., Werner F., 1997. Stability of an optimal schedule in a job shop', *Omega. International Journal of Management Sciences,* 25, 397-414.
8. Sotskov Yu.N., Tanaev V.S., Werner F., 1998. Stability radius of an optimal schedule: A survey and recent developments, Chapter in: G. Yu (Ed.), *Industrial Applications of Combinatorial Optimization,* 16, Kluwer Academic Publishers, Boston, MA, 72-108.

This page intentionally left blank

Chapter 11

SIMPLE PLANT LOCATION PROBLEM WITH REVERSE FLOWS

Zhiqiang Lu, Nathalie Bostel, Pierre Dejax

Abstract: We analyze the characteristics of the logistics systems including both direct and reverse flows, their activities and structure. We address the strategic planning problem of such systems and propose a facility location model as an extension of the simple plant location problem while taking into account the specific constraints related to reverse flows of directly reusable items. For solving this model, an algorithm based on lagrangian heuristics is developed. Numerical experiments are presented and the influence of reverse flows on the number and location of facilities is discussed. Extensions to this research are also proposed.

Key words: logistics, reverse logistics, facility location, lagrangian relaxation.

1. BASIC CONCEPTS OF REVERSE LOGISTICS

Complying with the changes in legislation, and protecting environment as well as economic and service reasons, more and more enterprises now take into account the reverse flows going backwards from customers to manufacturing plants or distribution centers within their logistics systems [17, 8] and a new domain is emerged - Reverse Logistics (RL). In fact, reverse logistics is not only the requirement of mitigating the burden on the environment, but a measure of improving the competence of enterprises and customer service level, and reducing the production costs [12]. However, little research work has been up to now devoted to this domain, particularly to the planning and optimization of reverse logistics systems.

We define reverse logistics as follows: "Reverse logistics can be viewed as an evolution of traditional forward logistics under environmentally-conscious industry or by other commercial drives. It encompasses all the logistics activities and management functions for reintroducing valued-objects, which finish or are not suitable to perform its primary function any more, into recovery systems for either recapturing their value or proper disposal".

A RL system comprises a series of activities, which form a continuous process to treat return-products until they are properly recovered or disposed of. According to industrial practices, a RL network may include some or all of the following activities: collection, cleaning, disassembly, test and sorting, storage, transport, and recovery operations. And this last activity can also be represented as one or a combination of several main forms, like reuse, repair, refurbishing, remanufacturing, cannibalization and recycling [2, 21, 6].

Figure 11-1 proposes a general representation of logistics systems with reverse flows, in which we can find two opposite directions of flows: the forward flow, consisting of initial products, flow from producers to customers and the reverse flow, constituted of the return-objects, from customers to recovery centers. Possible activities/stages concerning the forward and reverse channels have also been indicated in such a combined system.

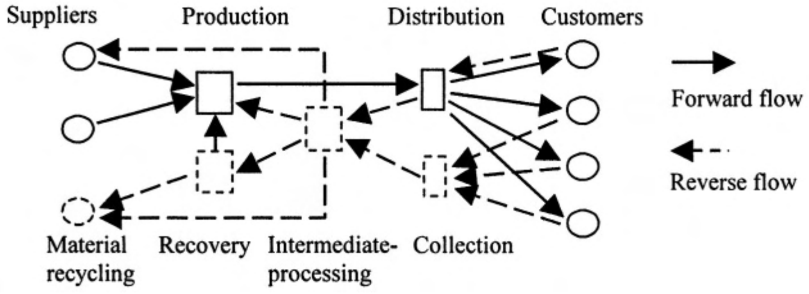


Figure 11-1. A framework of logistics systems with reverse flows

For the purpose of planning system activities, two elements are very important in order to facilitate the identification of the system characteristics: the type of return items and the main options adopted by the system for recovery. According to [7, 22], three principal types of return-items can be distinguished: packages, rotable spare parts and consumer goods; and four principal options of recovery can be categorized: reuse, repair, recycling and remanufacturing. Based on these two elements, four kinds of typical RL networks can be classified as follows:

- Directly Reusable Network (DRN). The return-objects consisting of the reverse flows in this network can be directly reused or need only a little reprocessing, such as cleaning or minor maintenance, to produce new products or to be used again as necessary equipment for transport. Examples of this kind of returns are pallets, bottles and containers.

- Remanufacturing Network (RMN). The purpose of such a system is to remanufacture parts or components to become new again and to be included in new products. The return-objects can be used products or consumable goods that are at the end of their lifetime and are sent back for recovery.
- Repair Service Network (RSN). The objective of such a network is to satisfy the service requirement of customers for the repair of defective products. The return-objects can be rotable parts of durable products, which are returned upon failure or for preventive maintenance.
- Recycling Network (RN). This kind of network system is found mainly for recycling raw materials. The return-objects can be used products that have no value per se and will be recycled in the form of raw materials.

Each of these types of RL systems is characterized by specificities. For this reason, different design and planning approaches have to be adapted for each case. Next, we discuss the facility location problem of RL systems.

2. FACILITY LOCATION PROBLEMS FOR REVERSE LOGISTICS SYSTEMS

In strategic decisions, a basic question involves the location and sizing of the main facilities for a logistics system. An important literature devoted to applications, models and algorithms for the location problem can be found. See for example [4, 10, 5].

Generally, because of the presence of recovery activities (recovering/reusing used products or materials), reverse logistics imposes some new characteristics on the management of the logistics system [6]. It is recognized that the system network of reverse logistics is not usually the symmetrical image of the traditional network [7]. This means that new functions or new actors can be introduced into the system, e.g. the implementation of back-shipments of reusable materials, or the location of collection, testing and sorting, or recovery centers. Fleischmann *et al.* [7] discusses the new issues that arise in this context of reverse logistics, reviews the mathematical models proposed in the literature, and finally points out the needs for further research.

In the published literature, most of studies propose to extend classic facility location models (Mixed Integer Programming) to support the analysis of location decision for RL systems. According to whether both forward and reverse activities (channels) are simultaneously covered in the considered system, the models can be classified into two categories:

independent models and integrated models. Only reverse channel is considered in the independent models and such works can be found as [1, 11,9, 18, 19]. For the integrated models, both forward and reverse channels are included, for example, the models proposed in [3, 15, 8]. Even in the integrated models, different relationships between forward channel and reverse channel can be formulated, for example in [13], the weak and strong relationships of correlation are distinguished. The presence of these relationships requires to introduce new constraints into the modeling of RL systems.

It should be noted that few research up to now considers simultaneously the forward and reverse activities in one single system and study their mutual constraints. In fact, in some situations and particularly in closed–loop systems, such interactions exist and need to be taken into account [7], for example in DRN and RMN systems [13]. Next, as an example we study the facility location problem for Directly Reusable Network (DRN), in which two types of flows are included.

3. A FACILITY LOCATION MODEL FOR DIRECTLY REUSABLE NETWORK

For a Directly Reusable Network, we assume there are two kinds of actors, i.e. producers and distributors (or customers) (see Figure 11-2). The producers provide (forward) products to distributors to satisfy their demands. At the same time, some reusable materials, e.g. containers, bottles, pallets or packages, to which we give the generic name of “containers”, need to be shipped back from distributors to producers for reutilization, on the grounds of economic and/or environmental considerations. By definition of this case (DRN), all of the returns to the production sites will be directly reused in the process of forward production or transportation. They may, however, only need a little reprocessing, such as cleaning or minor maintenance. This may be performed in a recovery center at the site of the producer.

We make the following assumptions. All demands for products and available return-items at the distributors are assumed to be given and deterministic. The product demands of distributors can be satisfied by any of the producers, and return-items at distributor points can be transported back to any of the recovery centers, attached to each of the producers. In our case, return-items are directly related to the production or shipment processes of “forward” products. In other words, at each of the producer recovery centers, sufficient quantities of reusable items are indispensable to accomplish the required operations of production or shipment from this producer. However, in the case of a lack of reusable items at any site of the producers, the necessary “containers” can be bought at a cost. We assume that, at each site of the producers, there is enough capacity for reprocessing return-items and the corresponding costs can be ignored.

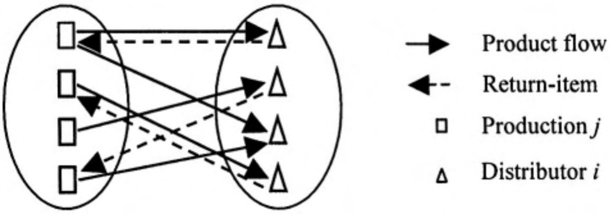


Figure 11-2. Directly Reusable Network

3.1 Formulation

We introduce the following notation:

Parameters:

$j \in J = \{\ 1,2,\dots,m\ \}$, index of potential location sites for production/recovery facilities,

$i \in I = \{\ 1,2,\dots,n\ \}$, index of distributor sites,

f_j = total fixed costs of setting up a production facility and recovery center at site *j*,

h_i = demand for products at distributor site *i*,

c_{ij} = total cost for satisfying a unit demand of distributor site *i* by potential (producer) site *j* (including costs of production and transportation),

hr_i = available quantity of return-items at site *i*,

cr_{ij} = total cost of recovery and transportation for unit return-item from site *i* to potential producer site *j*,

cb = unit cost for obtaining new “containers” at any location of producer,

γ = number of (forward) products enclosed in one “container” unit.

Decision variables:

$$Y_j = \begin{cases} 1, & \text{if a facility is located and set up at potential site } j, \\ 0, & \text{otherwise,} \end{cases}$$

X_{ij} = fraction of product demand at site *i* that is served by a producer at site *j*,

XR_{ij} = fraction of available quantity of return-items at site *i* that is taken back to site *j*,

XB_j = quantity of “containers” obtained externally at site *j*.

We formulate our problem as an uncapacitated facility location model, named *DRNU,* for the direct reusable network (DRN):

$$Min \sum_j f_j \; Y_j + \sum_i \sum_j c_{ij} \; h_i \; X_{ij} + \sum_i \sum_j cr_{ij} \; hr_i \; XR_{ij} + \sum_j cb \; XB_j \; , \qquad DRNU$$

s.t.

$$\sum_j X_{ij} = 1, \qquad \forall i, \qquad (1)$$

$$\sum_j XR_{ij} = 1 \;, \qquad \forall i, \qquad (2)$$

$$\sum_i hr_i \; XR_{ij} + XB_j \geq (1/\gamma) \sum_i h_i \; X_{ij} \;, \qquad \forall j, \qquad (3)$$

$$X_{ij} \leq Y_j, \qquad \forall ij, \qquad (4)$$

$$XR_{ij} \leq Y_j, \qquad \forall ij, \qquad (5)$$

$$Y_j \in \{0,1\}, \qquad \forall j, \qquad (6)$$

$$X_{ij}, XR_{ij} \geq 0, \qquad \forall ij, \qquad (7)$$

$$0 \leq XB_j \leq M \; Y_j, \qquad \forall j, \qquad (8)$$

M: number large enough.

Solving the model will allow the selection of facility locations while considering simultaneously forward and reverse flows and will thus examine the impact of both types of flows on location-allocation decisions. The objective of the model is to minimize the total cost of the system. Constraints (1) and (2) stipulate respectively that the demands for products and return-items must be fully served. Constraints (3) represent the relationship between the forward flows and return flows at each site *j*, that is the available quantity of containers (return-items) at site *j* should be proportionally greater than the quantity of products which will be produced or transported from the node *j*. Constraints (4) and (5) link the location and allocation variables, and constraints (6) specify the integrality of the location variables. Constraints (7) are non-negative constraints. Constraints (8) specify that the external provision of containers is only possible at sites *j* which are open. Constraints (2) and (5) are the reciprocal parts of constraints (1) and (4) for the return flows, while constraints (3) are key constraints of this model, linking forward and reverse flows.

Before we discuss our approach to solve the problem, some simplifications of the above model are studied. Without loss of generality for the case of directly reusable network, we suppose that the unit external provision cost of "containers" at producer sites does not depend on the sites and is much greater than any total unit recovery cost of return-items from distributor sites, that is $cb >> cr_{ij}, \forall ij$. Thus, we can simplify constraints (3) and remove constraints (8) from the model. We also assume total needed "containers" in the system is not less than availability quantities of return-items of the system, i.e. $\sum_i hr_i \le (1/\gamma)\sum_i h_i$ (this is often the case for DRN, and even in the contrary case it is not difficult to derive the corresponding model and its algorithm from our following discussions). As a result, we introduce alternative constraints (3') and (3'a) described below to replace constraints (3) and (8).

$$\sum_i hr_i XR_{ij} \le (1/\gamma)\sum_i h_i X_{ij}, \qquad \forall j, \qquad (3')$$

$$XB_j = (1/\gamma)\sum_i h_i X_{ij} - \sum_i hr_i XR_{ij}, \qquad \forall j. \qquad (3'a)$$

In addition, the term $\sum_j cb\, XB_j$ in the objective function of the model will become constant, i.e. $cb((1/\gamma)\sum_i h_i - \sum_i hr_i)$. In summary, according to the above discussion, variables $XB_j, \forall j$ have no influence on the optimization, and their values can be determined after solving the problem. Thus, we transform the uncapacitated model *DRNU* into *DRNU'* which does not consider XB_j explicitly:

$$Min \sum_j f_j\, Y_j + \sum_i \sum_j c_{ij}\; h_i\; X_{ij} + \sum_i \sum_j cr_{ij}\; hr_i\; XR_{ij}\;, \qquad DRNU'$$

$$s.t.\; (1), (2), (4), (5), (6), (7),$$

$$\sum_i hr_i\; XR_{ij} \le (1/\gamma) \sum_i h_i\; X_{ij}\;, \qquad \forall j. \qquad (3')$$

Note that, in the case when the provision cost *cb* depends significantly on the location, the only changes in the model are the values of the coefficients of variables X_{ij} and $XR_{ij}, \forall ij$ in the objective function, which does not affect the validity of our model and subsequent algorithm.

3.2 An algorithm based on Lagrangian heuristics to solve model *DRNU'*

As a mixed integer programming model, the model *DRNU'* described above can be solved using a standard solver. This is particularly true for small to medium size problems. However, for large size problems specific solution techniques are more appropriate.

We propose a procedure and algorithm based on Lagrangian heuristics for solving model *DRNU'*. For a general description of the Lagrangian heuristic approach, we refer to [16, 4, 20, 23]. Next, we first explain how to obtain a lower bound evaluation and an upper bound evaluation of the objective function and then we describe the general procedure of the Lagrangian heuristics. The more detailed discussion can be found in [13, 14].

3.2.1 Lower bound evaluation

The solution of the Lagrangian relaxed problem often provides a good lower bound for the original problem. We relax demand constraints (1) and (2) of model *DRNU'* and associate respectively multipliers μ_i and φ_i in order to obtain the relaxed problem *L:*

$$L:$$

$$\underset{X,XR,Y}{Min} \sum_j f_j \, Y_j + \sum_i \sum_j \left(c_{ij} \, h_i + \mu_i\right) X_{ij} + \sum_i \sum_j \left(cr_{ij} \, hr_i + \varphi_i\right) XR_{ij} \; - \sum_i \mu_i - \sum_i \varphi_i \,,$$

$$s.t. \; (3'), (4), (5), (6), (7),$$

$$\mu_i, \varphi_i \; \textit{unrestricted}, \qquad \forall i. \qquad (9)$$

In each Lagrangian iteration, we need to solve the above problem L to provide a lower bound to the original problem under a set of given values of multipliers μ_i and φ_i. We note that this relaxed problem L can actually be decomposed into j sub-problems, one for each facility *j*. For any $j \in J$, we define problem LX_j:

$$LX_j : Min \sum_i (c_{ij} \, h_i + \mu_i) X_{ij} + \sum_i (cr_{ij} \, hr_i + \varphi_i) XR_{ij} + f_j,$$

$$s.t. \; (3'),$$

$$0 \le X_{ij}, XR_{ij} \le 1 \,, \qquad \textit{for given } j.$$

Let $X^*_{ij}(LX_j)$ and $XR^*_{ij}(LX_j)$ respectively represent the values of variables determined in the optimal solution of problem LX_j for each *j*, and let $v(LX_j)$ be the optimal value of the objective function. Problem L can be reformulated as:

$$L : Min \sum_j v(LX_j) Y_j - \sum_i \mu_i - \sum_i \varphi_i \,,$$

$$s.t.$$

$$Y_j \in \{0,1\}, \qquad \forall j.$$

Knowing the optimal values of problems $LX_j, \forall j$, problem L can be solved by simple inspection as follows:

$$\begin{array}{lll} Y_j = \begin{cases} 1, & \\ 0, & \end{cases} & \begin{array}{l} \text{if } v(LX_j) < 0, \\ \text{otherwise;} \end{array} & \forall j, \\ X_{ij} = X_{ij}^*(LX_j), & \text{if } Y_j = 1, & \forall j, \\ XR_{ij} = XR_{ij}^*(LX_j), & \text{if } Y_j = 1, & \forall j, \\ X_{ij} = XR_{ij} = 0, & \text{if } Y_j = 0, & \forall j. \end{array}$$

The optimal objective value of L can be evaluated as, $v(L) = \sum_j v(LX_j) Y_j^* - \sum_i \mu_i - \sum_i \varphi_i$, where Y_j^* is the optimal value of Y_j.

3.2.2 Solution to sub-problem LX_j

LX_j is a linear programming problem, but we must solve it for many times in each Lagrangian iteration. We develop below an algorithm to tackle this problem. We distinguish two cases. For a given *j*, let

$$\begin{array}{lll} V_i^x = c_{ij}\ h_i + \mu_i, & \forall i \in I, & I^x = \{\ i \mid\ V_i^x < 0, \forall i \in I\ \}, \\ V_i^{xr} = cr_{ij}\ hr_i + \varphi_i, & \forall i \in I, & I^{xr} = \{\ i \mid\ V_i^{xr} < 0, \forall i \in I\ \}. \end{array}$$

(1)Case 1: if $\sum_{i \in I^{xr}} hr_i \le (1/\gamma) \sum_{i \in I^x} h_i$.

We define problem LX_j^0 as a relaxed problem of LX_j without considering constraint (3'). Then, we can see LX_j^0 is a simple problem, and we can determine its optimal solution set XXR^0 immediately as follows:

$$X_{ij} = \begin{cases} 1, & \text{if } i \in I^x, \\ 0, & \text{otherwise;} \end{cases} \qquad XR_{ij} = \begin{cases} 1, & \text{if } i \in I^{xr}, \\ 0, & \text{otherwise.} \end{cases}$$

In fact, in case 1, it is obvious that the above solution XXR^0 is also the optimal solution of the original problem LX_j, because this solution also satisfies constraint (3'). Therefore, we have solved problem LX_j in case 1.

(2) Case 2 : if $\sum_{i \in I^{xr}} hr_i > (1/\gamma) \sum_{i \in I^x} h_i$.

Firstly, we claim a property for the optimal solution of LX_j below, and the demonstration can be referred to [13], and [14].

For problem LX_j *in case 2, there must exist an optimal solution* $XXR^* = \{\ X_{ij}^*, XR_{ij}^*, i \in I\ \}$, *which satisfies constraint (3') to equality, that is,* $\sum_i hr_i\ XR_{ij}^* = (1/\gamma) \sum_i h_i\ X_{ij}^*, i \in I,$ *for given* j.

Taking advantage of this property, we propose an algorithm to find an optimal solution of LX_j in case 2. Let us denote two series *CX* and *CXR*. *CX* is a set of normalized coefficients of *X* sorted in ascending order and defined as $CX = \{(V_i^x / (1/\gamma)h_i)_s, i \in I, s \in IS = \{1,2,\cdots\}\}$ such that,

$$(1/\gamma)h_i \neq 0 \text{ and } (\frac{V_i^x}{(1/\gamma)h_i})_s \leq (\frac{V_{i'}^x}{(1/\gamma)h_{i'}})_{s'},\ i,i' \in I, i \neq i'; \forall s,s' \in IS, s < s';$$

And define $CXR = \{(V_i^{xr} / hr_i)_t, i \in I, t \in IRS = \{1,2,\cdots\}\}$ such that,

$$hr_i \neq 0 \text{ and } (\frac{V_i^{xr}}{hr_i})_t \leq (\frac{V_{i'}^{xr}}{hr_{i'}})_{t'}, i,i' \in I, i \neq i'; \forall t,t' \in IRS, t < t';$$

The algorithm, which we proposed to problem LX_j in case 2, i.e. with constraint (3') to be equality, is as follows.

① Initialization. Set $X_{ij} \leftarrow 0, XR_{ij} \leftarrow 0, \textit{for all } i \in I$;

For any $i \in I$ s.t. $V_i^x < 0$ and $(1/\gamma)h_i = 0$, set $X_{ij} \leftarrow 1$; And for any $i \in I$ s.t. $V_i^{xr} < 0$ and $hr_i = 0$, set $XR_{ij} \leftarrow 1$.

② Construct series *CX* and *CXR* according to our definition above. (It concerns a procedure to sort the given expressions in ascending order).

③ Generate the optimal solution, where $|IS|$ and $|IRS|$ are respectively the cardinalities of sets *IS* and *IRS*, and $((1/\gamma)h_i)_s$, $(hr_i)_t$, $(X_{ij})_s$, and $(XR_{ij})_t$ are respectively the corresponding $(1/\gamma)h_i, hr_i, X_{ij}, XR_{ij}$ included in the *s*th and *t*th terms in series *CX* and *CXR*.

```
BEGIN
SET $s \leftarrow 0$; $t \leftarrow 0$; $val \leftarrow 0$; $valR \leftarrow 0$; $valMin \leftarrow 0$;
WHILE $s < |IS|$ and $t < |IRS|$
   IF $val = 0$ THEN $val \leftarrow ((1/\gamma)h_i)_s$ ENDIF
   IF $valR = 0$ THEN $valR \leftarrow (hr_i)_t$ ENDIF
     IF $(V_i^x / (1/\gamma)h_i)_s + (V_i^{xr} / hr_i)_t < 0$ THEN
     SET $valMin \leftarrow Min(val, valR)$;
     SET $val \leftarrow val - valMin$; $valR \leftarrow valR - valMin$;
     IF $val = 0$ THEN $(X_{ij})_s \leftarrow 1$, and $s \leftarrow s+1$; ENDIF
     IF $valR = 0$ THEN $(XR_{ij})_t \leftarrow 1$, and $t \leftarrow t+1$; ENDIF
  ELSE break;
  ENDIF
ENDWHILE
IF $s < |IS|$, $val \neq 0$ and $val \neq ((1/\gamma)h_i)_s$ THEN $(X_{ij})_s \leftarrow \frac{((1/\gamma)h_i)_s - val}{((1/\gamma)h_i)_s}$ ENDIF
```

```
    IF $t < |IRS|, valR \neq 0$ and $valR \neq (hr_i)_t$ THEN $(XR_{ij})_t \leftarrow \frac{(hr_i)_t - valR}{(hr_i)_t}$ ENDIF
END
```

In case 2, the above procedure yields a solution to LX_j with a minimal objective value while keeping constraints (3') satisfied to equality and thus this solution is an optimal solution to LX_j. This procedure is an assignment process based on two sorted series of *CX* and *CXR*, and its complexity is of $O(n)$. Additionally, the complexity of sorting procedure can be classically of $O(n^2)$ (in fact, at best $O(n \lg n)$).

We have therefore solved the relaxed problem *DRNU'*, and its optimal objective value provides us with a lower bound at each Lagrangian iteration for a set of given values of μ_i and φ_i. The lower bound *lb* is given by *v(L).*

3.2.3 Upper bound evaluation

For each Lagrangian iteration, we can obtain the optimal values of the location variables $Y_j^*(L)$ for problem *L,* which are determined by solving the relaxed problem *L* (see 3.2.1). Using these values, a feasible solution of the original problem *DRNU'* can possibly be found, and this solution can be used as an upper bound. Actually, if all the variables of Y_j have been determined, the original problem *DRNU'* becomes a transportation problem i.e. $T = DRNU'(X, XR \mid \hat{Y})$, $\hat{Y}$ is the vector of $Y_j^*(L)$, $j \in J$. In the case where there does not exist the feasible solution with these values, we simply do not determine the upper bound at this iteration.

Let $v(T)$ be the optimal value of problem *T.* We then determine the upper bound $ub = \sum_j f_j Y_j^*(L) + v(T)$.

3.2.4 General procedure for Lagrangian heuristics

The Lagrangian procedure requires the updating of multipliers at each iteration for the continuous improvement of the lower bound and upper bound. In order to update the Lagrangian multipliers μ_i and φ_i, the subgradient optimization method is employed. The updated values of these multipliers can be formulated as follows (*p* is the iteration number; the variables with superscripts *p* or *p*+1 represent their values at the *p*th *or p*+1th iteration):

$$\mu_i^{p+1} = \mu_i^p + \frac{\alpha^p (UB - lb^p)(\sum_j X_{ij}^p - 1)}{\sum_i ((\sum_j X_{ij}^p - 1)^2 + (\sum_j XR_{ij}^p - 1)^2)},$$

$$\varphi_i^{p+1} = \varphi_i^p + \frac{\alpha^p (UB - lb^p)(\sum_j XR_{ij}^p - 1)}{\sum_i ((\sum_j X_{ij}^p - 1)^2 + (\sum_j XR_{ij}^p - 1)^2)}.$$

UB is the best upper bound found so far at the current iteration. lb^p is the value of the lower bound at the *p*th iteration; α^p is a positive parameter, and its value is generally set to be equal to or less than 2. During the iterations, we decrease its value whenever the best lower bound has not been improved over a fixed number of iterations. The general procedure of the Lagrangian heuristic algorithm to solve *DRNU'* can be described as follows:

Step 1. (Initialization) Set initial values for the Lagrangian multipliers μ_i and φ_i (values with the same order of magnitude as $c_{ij}h_i$ and $cr_{ij}hr_i$ will do). Set the initial value of *UB* equal to the value of the objective function for any feasible solution to *DRNU'* (for example, choosing any single facility to be open will do –see Step 3).

Step 2. Using the algorithm proposed above for solving the Lagrangian relaxed problem *L,* calculate the current lower bound lb^p , and determine the corresponding location variables. Update the best lower bound *LB* if possible.

Step 3. By solving the transportation problem *T* with the location values obtained in Step 2, calculate a feasible solution of the original problem as an upper bound. Update the best upper bound *UB* if possible.

Step 4. Update the Lagrangian multipliers using the subgradient method (as indicated above in this section).

Step 5. If any terminating condition has been met, stop, otherwise go back to Step 2.

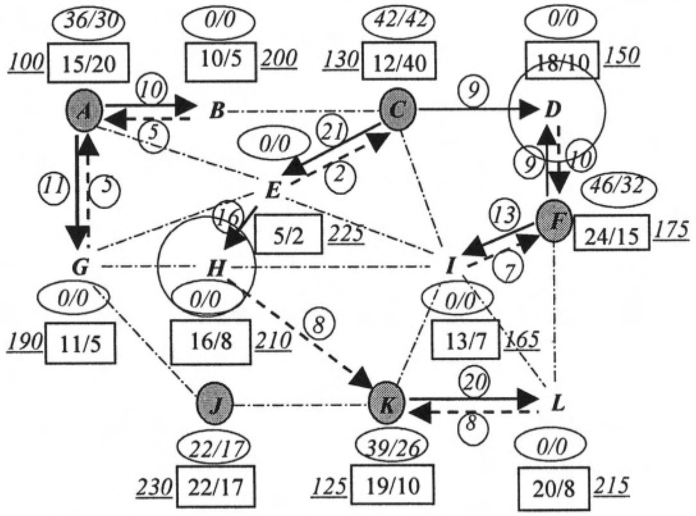
The criterion for termination is one of three conditions: ① A specified time limit has been reached; ② The gap between lower bound and upper bound is close enough, i.e. $(UB - LB)/LB < \varepsilon$, where ε is a specified parameter (we fix ε equal to 1%); ③ The step size becomes too small to allow any improvement of the lower bound.

3.2.5 Numerical experiments

We have implemented the above algorithm using C++, and the linear program solver is Cplex Solver 6.5. All test examples presented below are run on a PC with processor Intel Pentium 4, 1300MHz, and 524MB memory.

For illustration, we have adapted a small example (example_1) from [4] by allocating available quantities of return-items to distributor sites (see Figure 11-3). In this example, there are 12 potential location sites both for production and recovery centers. The distances between the nodes in the network are identical as those in [4]. The unit cost per demand-distance for forward products is 0.35, and for return items it is supposed to be 90% of the unit cost for forward products.

At the optimum, the selected locations and optimized flows for the network considering reverse flows are indicated in Figure 11-3. The objective function value is 1604.93 (not including the cost of "containers" obtained externally). Five facilities will be set up at sites A, C, F, J, K. In [4], the optimal result for the location problem of this network without reverse flows is given with an objective value of 1234.95 and three selected facilities at nodes A, D, K.



Notes for figure:

① *A*, *B*, ..., *L* are the potential facility locations as well as the distributor sites; Numbers in the boxes are respectively the demand for products and the available quantity of return-items. Numbers underlined beside the boxes are the fixed costs for the nodes.

② Numbers circled above the boxes are the production volume and the recovered volume determined after optimization. For example, at site A, the production level has been determined at 36, and the quantity of return items received at 30.

③ Numbers circled above the lines are the transportation volumes. The directed solid lines represent the forward flows and the directed dashed lines represent the reverse flows.

④ Letters in dark circles represent selected sites for production/recovery.

Figure 11-3. The result of optimization on a small example

From the results, we can see that the introduction of reverse flows influences the whole network design. In our case, it changes both the facility locations and flow allocations. Of course, the degree of influence is dependent on the relative magnitude of forward and reverse flows and their respective costs. In the case where an appropriate volume of return-items is indispensable to the forward flows, like in our situation, the allocation of return-items to a suitable recovery center is also important and will affect the total cost of the system. An interesting point to make relates to the allocation of customers to facilities. For the traditional uncapacitated facility location model, the demand of each node will be totally allocated to a single selected facility. This may, however, not always be the case in networks with return flows. In our example, the demand of site D will be supplied by both nodes C and F (Figure 11-3). Such a result is caused by the mutual restrictions between forward and reverse flows, which is represented by coupling constraint (3'). This is a key difference between the classical facility location model and the model combining forward and reverse flows. Additionally, it is not necessary to have common assignments for demands of "forward" products and "reverse" return-items at one node. In Figure 11-3, the demand of products at D is served by both C and F, but the return-items at D is only returned back to F; at node H, the demand of products and return-items are respectively satisfied by C and K.

We now present another experimental example (example 2) consisting of a large problem that is adapted from the OR-Library: http://www.ms.ic.ac.uk/jeb/orlib/. It contains 25 candidate location sites and 50 customers. The fixed costs for each facility of production and recovery is 12 500 except for the 11th candidate site which is without fixed cost. The demand weighted distances between candidate location sites and customers are given on this website. Costs per demand-distance are set to 1 and the coefficient of transport costs for return-items is 0.9.

Table 11-1 shows the several results that we obtained using the algorithm described above. We have found that the algorithm provides a good performance in terms of gaps between *LB* and *UB* as well as computing time. The calculation processes for all of the cases terminate under terminating condition ② (see 3.2.4). In Table 11-1, we present the results for different return-rates, going from 10% to 100%, as well as the case without any return flows. These results demonstrate the influence of the reverse flows on the number and sites of facilities. The column "locations" indicates the numbers of selected facilities and the list of such facilities.

4. CONCLUSIONS

Although some of the concepts of reverse logistics, such as the recycling of products, have been put into practice for years, it is only fairly recently that the integration of reverse logistics activities has been a real concern for the management and organization of logistics systems. We introduce the basic concepts of reverse logistics and four kinds of typical reverse logistics networks. Concentrating on the strategic problem of the design of logistics systems, as an example, we propose a facility location model in one particular case of Directly Reusable Network, in which both "forward" and "reverse" flows are simultaneously considered. To solve the problem, we propose an algorithm based on the Lagrangian heuristic approach as an alternative to using a standard solver. Numerical tests are conducted on data adapted from classical test problems, showing a good performance of the algorithm. Analyzing the results, we show that reverse flows influence the decisions about location and the allocation of flows.

This research has been extended to the problems of facility location for the other types of reverse logistics systems that we have described before [13]. It can also be included in the general framework of the hierarchical planning of logistics systems with reverse activities.

Table 11-1. Summary of the numerical experimental results on example 2

Reverse flows (return rate)	*Objective values*	*Transport Costs*	*Locations*	*LB*	*UB*	*Time*
10%	920378	795378	1,4,6,7,11,12, 13,17,23,24,25	915354	920378	5s
30%	1054060	929060	1,4,6,7,9,11, 13,17,23,24,25	1045450	1054060	8s
50%	1177510	1015010	1,2,4,6,7,9,11,12, 13,17,18,23,24,25	1168260	1177510	5s
80%	1364640	1189640	1,2,4,6,7,8,9,11,13, 17,18,20,23,24,25	1354810	1364640	6s
100%	1489160	1301660	1,2,4,6,7,8,9,11,13, 17,18,20,21,23,24,25	1476070	1489160	5s
Without returns	855467	742967	1,4,6,7,11, 13,17,23,24,25	849324	855467	10s

REFERENCES

1. Barros A.I., Dekker R., Scholten V., 1998. A two-level network for recycling sand: a case study, *European Journal of Operational Research,* 110, 199-214.
2. Beaulieu M, Martin R., Landry S., 1999. Logistique à rebours: un portrait nord-américain, Logistique & Management, 7, 5-14.
3. Bloemhof-Ruwaard J. M., Salomon M., Van Wassenhove L. N., 1996. The capacitated distribution and waste disposal problem, *European Journal of Operational Research,* 88, 490-503.
4. Daskin M. S., 1995. *Network and discrete location, models, algorithms, and application,* Wiley-Interscience Publication, John Wiley and Sons.
5. Dejax P., 2001. Stratégie, planification et implantation du système logistique, in J.P. Campagne et P.Burlat (eds.), *Maîtrise et organisation des flux industriels,* Hermes, Lavoisier, 129-160.

6. Dekker R., Van Der Laan E.A., 1999. Gestion des stocks pour la fabrication et la refabrication simultanées: synthèse de résultats récents, *Logistique & Management,* 7, 59-64.
7. Fleischmann M., J. M. Bloemhof-Ruwaard, R. Dekker, E. Van Der Laan, J. A.E.E. Van Nunen, L.N. Van Wassenhove, 1997. Invited review, quantitative models for reverse logistics: a review, *European Journal of Operational Research,* 103, 1-17.
8. Fleischmann M., Beullens P., J. M. Bloemhof-Ruwaard and L. N. Van Wassenhove, 2000. The Impact of Product Recovery on Logistics Network Design, *working paper of the Center for Integrated Manufacturing and Service Operations,* INSEAD, 2000/33/TM/CIMSO 11.
9. Jayaraman V., VDR Guide Jr, R. Srivastava, 1999. A closed-loop logistics model for remanufacturing, *Journal of the Operational Research Society,* 50, 497-p508.
10. Labbé M., Louveaux F., 1997. Location problems, in M. Dell'Amico, F. Maffioli and S. Martello (eds.), *Annotated Bibliographies in Combinatorial Optimization,* J.Wiley and Sons, 261-282.
11. Louwers D., Kip B.J., Peters E., Souren F., Flapper S.D.P., 1999. A facility location allocation model for reusing carpet materials, *Computer & Industrial Engineering,* 36, 855-869.
12. Lu Z., Bostel N., Dejax P., 2001. Planification hiérarchisée des systèmes logistiques incluant la logistique inverse: Problématique et modèles stratégiques, *Actes du 4e congrès international de génie industriel* (GI2001), Aix-en-Provence-Marseille, France, 1141-1151.
13. Lu Z., 2003a. Hierarchical planning and optimization of logistics systems with reverse flows, *Ph.D thesis of Université de Nantes,* France.
14. Lu Z., Bostel N., Dejax P., 2003b. A facility location model and algorithm for the planning of reverse logistic systems in the case of directly reusable items, *Rapport de Recherche of Ecole des Mines de Nantes,* France, 03/4/AUTO.
15. Marín A., Pelegrín B., 1998. The return plant location problem: modeling and resolution, *European Journal of Operational Research,* 104, 375-392.
16. Parker R.G., Rardin R.L., 1988. *Discrete Optimization,* Academic Press, Inc.
17. Rogers D.S., Tibben-Lembke R.S., 1998. *Going backwards: reverse logistics trends and practices,* Center for Logistics Management, University of Nevada, Reno, Reverse Logistics Executive Council.
18. Shih L., 2001. Reverse logistics system planning for recycling electrical appliances and computers in Taiwan, Resources, *Conservation and Recycling,* 32, 55-72.
19. Spengler Th., Püchert H., Penkuhn T., Rentz O., 1997. Environmental integrated production and recycling management, *European Journal of Operational Research,* 97, 308-326.
20. Sridharan R., 1995. The capacitated plant location problem, *European Journal of Operational Research,* 87, 203-213.
21. Thierry M.C., Salomon M., Van Nunen J.A.E.E., Van Wassenhove L.N., 1993. Strategic Production and Operations Management Issues in Product Recovery Management, *Management Report Series,* No. 145, Erasmus Universiteit/Rotterdam School of Management.
22. Thierry M.C., Salomon M., Van Nunen J., Van Wassenhove L., 1995. Strategic issues in product recovery management, *California Management Review* 37, 114-135.
23. Tragantalerngsak S., Holt J., Rönnqvist M., 1997. Lagrangian heuristics for the two-echelon, single-source, capacitated facility location problem, *European Journal of Operational Research* 102, 611-625.

Chapter 12

CONCAVE COST SUPPLY MANAGEMENT FOR SINGLE MANUFACTURING UNIT

Satyaveer Singh Chauhan, Anton Eremeev,
Alexander Kolokolov, Vladimir Servakh

Abstract: The considered problem consists of product delivery from a set of providers to the manufacturing units (single unit and single planning period in our case). The cost function is concave. Given the lower and upper bounds on the shipment size for each provider, the demand of the manufacturing unit has to be satisfied. In this chapter it is shown that this optimisation problem is *NP*-hard even to find a feasible solution to this problem. Considering the problem in integer programming formulation we propose a pseudo-polynomial algorithm, using the dynamic programming technique. Some possible approaches to solving the problem with multiple manufacturing units are discussed.

Key words: concave cost, supply, dynamic programming, integer programming, complexity.

1. INTRODUCTION

In this chapter, we consider the problem where a set of providers supply one type of product to a manufacturing unit, the quantity that can be delivered lies between the given minimum and maximum values, and the costs proposed by each provider are concave functions of quantity being delivered. The concavity assumption reflects a common situation taking place in industry since usually the unit cost of products and the transportation unit cost decrease as the size of an order increases. This problem is similar to, but different from the well known transportation problem with concave costs (see e.g. [1, 8]), but in our case a provider either delivers a quantity of product that lies between a lower bound and an upper bound or delivers nothing. The lower bound is the economical production quantity imposed by the provider and the upper bound is a technical constraint: it is the maximum quantity the provider is able to produce during the period under consideration. Formally the problem is stated as follows:

$$\sum_{i=1}^{n} k_i(x_i) \to \min, \tag{1}$$

$$\sum_{i=1}^{n} x_i = A, \tag{2}$$

$$x_i \in \{0\} \cup [m_i, M_i] \text{ for } i = 1,2,\ldots,n. \tag{3}$$

Here n is the number of providers, x_i is the quantity of product delivered to the manufacturing unit from provider i; A is the total amount of product required for the manufacturing unit; m_i is the minimum quantity the provider i is prepared to deliver due to the economical reasons; M_i is the maximum quantity the provider i is able to deliver. All quantities of product here and in the rest of the chapter refer to some standard planning period (e.g. one week). The cost $k_i(x_i)$ is

$$k_i(x_i) = \begin{cases} 0 & \text{if} \quad x_i = 0, \\ a_i + g_i(x_i) & \text{if} \quad x_i > 0, \end{cases}$$

where $a_i \geq 0$ and $g_i(x_i) \geq 0$ are concave and non-decreasing functions when x_i is positive, *i=1,...,n.* This problem formulation was suggested in [2] not only for the single manufacturing unit but also for the general case with multiple manufacturers (see the further discussion in Section 3). A number of useful properties of the problem have been shown and several heuristic algorithms were proposed and tested there. Our goal here is to investigate the exact solution methods and complexity issues of the problem with single manufacturing unit and some of its extensions.

Firstly, in Section 2 we demonstrate the *NP*-hardness of the problem and show that the standard dynamic programming approach allows to find the optimum in pseudo-polynomal time. A discussion on extension of exact solution methods for the case of several providers, and the conclusions are contained in Sections 3 and 4.

2. PROBLEM COMPLEXITY AND PSEUDO-POLYNOMIAL TIME ALGORITHM

Theorem 1. *Finding a feasible solution to supply management problem (1)-(3) with rational input parameters is NP-hard.*

Proof: Let there be a polynomial time algorithm which finds a feasible solution satisfying (2) and (3) when such solutions exist. Assume that $A = \frac{1}{2}\sum_{i=1}^{n} m_i$, M_i , m_i are integer and $M_i = m_i$ for all $i = 1, 2, ..., n$. By substitution $x_i = z_i \cdot m_i$, i = 1, 2,...,n. conditions (2) and (3) for this case may be written as follows:

$$\sum_{i=1}^{n} m_i z_i = \frac{1}{2}\sum_{i=1}^{n} m_i \ , \tag{4}$$

$$z_i \in \{0, 1\} \text{ for } i = 1, 2, ..., n \, . \tag{5}$$

The polynomial time algorithm mentioned above is suitable to recognize the consistency of (4) and (5) which is equivalent to solving the *NP*-complete SUBSET SUM problem [3]. Q.E.D.

In what follows we suppose that all A, m_i, M_i are integer, which is a certain limitation, nevertheless its influence may always be reduced by choosing the sufficiently fine-grained scale of the variables. In our analysis we will use a fact similar to Result 1 from [2], although here we do not require that functions $g_i(x)$ are continuously differentiable:

Theorem 2. *If problem (1)-(3) is solvable, then there exists an optimal solution* $X = \{x_1, x_2, ..., x_n\}$ *such that* $x_i = m_i$ *or* $x_i = M_i$ *or* $x_i = 0$ *for* i = 1,2,...,n, *except for at most one* $j \in \{1, 2, ..., n\}$ *for which* $m_j < x_j < M_j$.

Proof: Let $X^1 = \{x_1^1, x_2^1, ...x_n^1\}$ be an optimal solution to problem (1)-(3). Assume there exists a pair $i, j \in \{1, 2, ..., n\}$ such that $m_i < x_i^1 < M_i$ and $m_j < x_j^1 < M_j$. Firstly, consider the case when we have

$$k_i(x_i^1+\delta)-k_i(x_i^1) \leq k_j(x_j^1)-k_j(x_j^1-\delta), \tag{6}$$

where by definition $\delta = \min(M_i - x_i^1, x_j^1 - m_j)$. Then we can set:

$$x_i^2 = x_i^1 + \delta \, , \tag{7}$$

$$x_j^2 = x_j^1 - \delta \, . \tag{8}$$

Let us denote by X^2 the new solution obtained after replacing x_i^1 by x_i^2 and x_j^1 by x_j^2 in X^1. Then adding (7) and (8) we see that (2) and (3) still hold for X^2. Besides that, $k_i(x_i^2)+k_j(x_j^2)\le k_j(x_j^1)+k_i(x_i^1)$, so X^2 is optimal too.

Now, if (6) does not hold, analogously we can treat the case when

$$k_i(x_i^1)-k_i(x_i^1-\Delta)\ge k_j(x_j^1+\Delta)-k_j(x_j^1), \tag{9}$$

where by definition $\Delta=\min(M_j-x_j^1,x_i^1-m_i)$.

Finally, let us prove that other options are impossible, i.e. an assumption that both

$$k_i(x_i^1+\delta)-k_i(x_i^1) > k_j(x_j^1)-k_j(x_j^1-\delta), \tag{10}$$

$$k_i(x_i^1)-k_i(x_i^1-\Delta) < k_j(x_j^1+\Delta)-k_j(x_j^1), \tag{11}$$

hold, will lead to a contradiction. Indeed, since $k_j(x_j)$ is concave, so $\frac{\Delta+\delta}{\delta}k_j(x_j^1)\ge k_j(x_j^1+\Delta)+\frac{\Delta}{\delta}k_j(x_j^1-\delta)$, i.e. $\frac{\Delta}{\delta}(k_j(x_j^1)-k_j(x_j^1-\delta))\ge k_j(x_j^1+\Delta)$ - - $k_j(x_j^1)$, and by (10) we have:

$$k_i(x_i^1+\delta)-k_i(x_i^1) > \frac{\delta}{\Delta}(k_j(x_j^1+\Delta)-k_j(x_j^1)).$$

Combining this with (11) we conclude that:

$$\Delta k_i(x_i^1+\delta)+\delta k_i(x_i^1-\Delta) > (\Delta+\delta)k_i(x_i^1),$$

which implies a contradiction with concavity of $k_i(x_i)$.

Thus, either (6) or (9) must hold, and consequently we always have:

$$\sum_{i=1}^{n}k_i(x_i^2)\le\sum_{i=1}^{n}k_i(x_i^1).$$

Continuing the same process will lead to a solution X indicated in the statement of the theorem. Q.E.D.

Since A, m_i, M_i are integer, so by Theorem 2 there exists an optimal solution where all x_i, $i=1,2,...,n$ are integer also. Thus the original continuous problem can be considered as a discrete optimization problem. In our analysis of this problem we will use the standard dynamic programming technique. Let us consider all possible integer values of variable x_n.

1. If $x_n=0$, then $k_n(0)=0$, and the problem reduces to the following:

$$\sum_{i=1}^{i=n-1}k_i(x_i)\to\min,$$

$$\sum_{i=1}^{i=n-1} x_i = A \ ,$$

$$x_i \in \{0\} \cup [m_i, M_i] \text{ for } i\text{=1,2,...,}n\text{-1}.$$

Let $\varphi(p,a)$ denote the optimal objective function value for the problem:

$$\varphi(p,a) = \min_x \left\{ \sum_{i=1}^{p} k_i\,(x_i\,) \right\},$$

$$\sum_{i=1}^{p} x_i = a \ ,$$

$$x_i \in \{0\} \cup [m_i, M_i] \text{ for } i = 1,2,..., p\,.$$

According to this notation in case $x_n = 0$ we have $\varphi(n,A) = \varphi(n-1,A)$.

2. If we consider some fixed $m_n \le x_n \le M_n$ then the problem (1)-(3) reduces to:

$$\sum_{i=1}^{i=n-1} k_i\,(x_i\,) + k_n(x_n) = \varphi(n-1, A - x_n) + k_n(x_n) \to \min,$$

$$\sum_{i=1}^{i=n-1} x_i = A - x_n \ ,$$

$$x_i \in \{0\} \cup [m_i\ , M_i\ ] \text{ for } i = 1,2,..., n\text{-1}.$$

Thus, if the positive shipment x_n is fixed then we need to solve the problem for n-1 providers and the smaller amount of product remaining. Combining the cases 1 and 2 what we have to find is the value of x_n that minimizes the goal function:

$$\varphi(n,A) = \min\left\{ \varphi(n-1,A),\ \min_{m_n \le x_n \le M_n} (\varphi(n-1,A-x_n) + k_n(x_n)) \right\}.$$

To find $\varphi(n,A)$ here we need the solutions to all problems of dimension n-1, which may be computed recursively through the problem with n-2 variables, etc. Finally we have the general formula:

$$\varphi(p,a) = \min\left\{ \varphi(p-1,a),\ \min_{m_p \le x_p \le M_p ; x_p \le a} (\varphi(p-1,a-x_p) + k_p(x_p)) \right\},$$

$$p = 1,2,...,n;\ \ a = 0,1,2,..., A.$$

The computations with this formula are carried out through double loop: with $p = 1,2,...,n,$ and with $a = 0,1,2,...,A,$ assuming the initial conditions $\varphi(0,0) = 0,\ \varphi(0,a) = \infty,\ a = 1,2,...,A.$

Calculation of $\varphi(p,a)$ requires not more than $M_p - m_p + 2$ comparison operations. So the total number of comparisons for solving (l)-(3) is bounded by $A \cdot \sum_{p=1}^{n}(M_p - m_p + 2)$.

Thus, there exists a pseudo-polynomial time algorithm for solving this problem (here we imply that functions $k_i(x_i)$, i=1,2,...,n are computable in polynomial time). In fact if we divide the problem data input string into two parts: substring *s'* for encoding the functions $k_1, k_2...,k_n$ and substring *s* for the rest of the data, then even in case of unbounded growth of the values of numeric parameters encoded in *s'* the running time will remain polynomial in the length of *s*. Therefore we have

Theorem 3. *Let s be the input substring, encoding A,* $m_1, m_2,..., m_n, M_1, M_2,..., M_n$. *If functions* $k_i(x_i)$, i=1,2,...,n *are polynomial time computable in length of s for all* $0 \le x_i \le A$, *then there exists a pseudo-polynomial time algorithm (with respect to input substring s) solving problem (1)-(3).*

Note that the complete enumeration of solutions has the time complexity $O(n3^{n-1})$. If A is large and n is small, the complete enumeration method may be advantageous. However with bounded A, $m_1, m_2,..., m_n, M_1, M_2,..., M_n$ the running time of the dynamic programming will be smaller by an exponential factor.

3. SOME APPROACHES TO SOLVING MORE GENERAL PROBLEMS

It is interesting to consider the extension of the concave cost supply management problem to the case with multiple manufacturing units as it was formulated in [2]:

$$\sum_{i=1}^{n}\sum_{j=1}^{m} k_{ij}(x_{ij}) \rightarrow \min, \tag{12}$$

$$\sum_{i=1}^{n} x_{ij} = A_j,\ j = 1,2,...,m, \tag{13}$$

$$\sum_{j=1}^{m} x_{ij} \le M_i\ ,\ i = 1,2,...,n, \tag{14}$$

$$x_{ij} \in \{0\} \cup [m_{ij}, M_i] \text{ for } i = 1,2,...,n;\ j = 1,2,...,m. \tag{15}$$

Here n is the number of providers and m is the number of manufacturing units, x_{ij} is the quantity of product delivered to the manufacturing unit j from provider i; A_j is the total amount of product required for the manufacturing unity; m_{ij} is the minimum quantity the provider i is prepared to deliver to the manufacturing unity; M_i is the maximum quantity the provider i is able to deliver to the manufacturing units. The cost $k_{ij}(x_{ij})$ is

$$k_{ij}(x_{ij}) = \begin{cases} 0 & \text{if} \quad x_{ij} = 0 \\ a_{ij} + g_{ij}(x_{ij}) & \text{if} \quad x_{ij} > 0, \end{cases}$$

where $a_{ij} \geq 0$ and $g_{ij}(x_{ij}) \geq 0$ are concave and non-decreasing functions when x_{ij} is positive, $i=1,...,m, j=1,...,n$.

The necessary conditions of the optimum formulated in [2] for this problem permit the development of a dynamic programming approach similar to that described above. However the time and memory resources required by such an algorithm might present serious obstacles. In this connection it seems to be appropriate to use the piecewise linear approximations of the functions $k_{ij}(x_{ij})$, since then the problem can be formulated as an integer linear programming problem. For example in the case of linear costs $k_{ij}(x_{ij}) = c_{ij}x_{ij}$, $i = 1,...,m$, $j = 1,...,n$ introducing supplementary Boolean variables z_{ij} we obtain the following mixed-integer problem:

$$\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \to \min$$

$$\sum_{i=1}^{m} x_{ij} = A_j, \quad j = 1,...,n,$$

$$\sum_{j=1}^{n} x_{ij} \leq M_i, \quad i = 1,...,m,$$

$$\left.\begin{aligned} & x_{ij} \geq z_{ij}m_{ij} \\ & x_{ij} \leq z_{ij}M_i \\ & x_{ij} \geq 0 \\ & z_{ij} \in \{0,1\} \end{aligned}\right\} \quad i = 1,...,m, \; j = 1,...,n.$$

A number of approaches may be used for solving such problem: the branch and bound methods of Land and Doig type (see e.g. [7]), the Benders decomposition method and cutting plane algorithms (see e.g. [1,4]), L-class enumeration algorithms [5,6], etc. Note that a further generalization of problem (12)-(15) may be done through the assumption that the size of shipment x_{ij} belongs to a range consisting of several intervals for all i and j. The approaches mentioned above could be extended to this case as well.

4. CONCLUSIONS

The concave cost supply management problem with single manufacturing unit was shown to be *NP*-hard and a dynamic programming pseudo-polynomial time algorithm was suggested for it. The possible approaches to solving the more general problem with multiple manufacturing units were discussed.

We expect that the further research will be aimed at the elaboration of the solution methods discussed in Section 3 and their theoretical and experimental comparison. Another direction for the further research is the analysis of a problem with lower bounds on consumption of product instead of the exact conditions (2) and (13) assumed in this chapter. In such a modification (at least in the single-unit case) the feasible solution is easier to find and fast approximation algorithms are appropriate.

ACKNOWLEDGEMENTS

The research was supported by INTAS Grant 00-217. The authors would also like to thank J.-M. Proth, G.G. Zabudsky and L.A. Zaozerskaya for the discussions and helpful comments on this result.

REFERENCES

1. Bakhtin A.E., Kolokolov A.A. and Korobkova Z.B., 1978. *Discrete Problems of Production-Transportation Type,* Nauka, Novosibirsk. (in Russian).
2. Chauhan S.S. and Proth J.-M., 2003. The concave cost supply problem. *European Journal of Operational Research,* 148 (2), 374-383.
3. Garey M.R. and Johnson D.S., 1979. *Computers and Intractability. A Guide to the Theory of NP-Completeness,* W.H. Freeman and Company, San Francisco.
4. Hu T.C., 1970. *Integer Programming and Network Flows,* Addison-Wesley Pbl.
5. Kolokolov A.A., 1994. Decomposition algorithms for solving of some production-transportation problems. In: *Preprints of Triennial Symposium on Transportation Analysis II.,* Vol. 1, Capri, Italy, 179-183.
6. Kolokolov A.A. and Levanova T.V., 1996. Decomposition and *L*-class enumeration algorithms for solving some location problems. *Vestnik Omskogo Universiteta,* 1, Omsk, OmSU, 21-23. (in Russian).
7. Schrijver A., 1986. *Theory of Linear and Integer Programming,* John Wiley Sons, Vol. 2.
8. Yan S. and Luo S.-C., 1999. Probabilistic local search algorithms for concave cost transportation network problems. *European Journal of Operational Research,* 117, 511-521.

Chapter 13

PRODUCT FAMILY AND SUPPLY CHAIN DESIGN

An integration of both points of view

Jacques Lamothe, Khaled Hadj-Hamou, Michel Aldanondo

Abstract: When designing a new family of products, designers and manufacturers must define the product family and its supply chain simultaneously. At the very first step of the design process, designers propose solutions of product decompositions. The second step is to select some of these decompositions while choosing the architecture of the supply chain. A mixed integer linear programming model is investigated that optimizes the operating cost of the resulting supply chain while choosing the product decompositions. This work is applied to an industrial problem of an automotive supplier.

Key words: product design, supply chain design, mixed integer linear programming.

1. INTRODUCTION

Nowadays, the growing demand for customizable or configurable products involves an increasing number of product variants and a growing complexity of products while controlling the products costs and the customer lead time. Consequently, when designing a new product family, a consistent approach is necessary to quickly define the family and its supply chain in order to guaranty the customer's satisfaction and to minimize the supply chain global operating costs.

We suggest a design approach that simultaneously defines a product family and its supply chain while facing a demand with a large diversity. Between product design and supply chain management, this approach is closely related to the field of Concurrent Engineering [20] and a "Design for Supply Chain" approach [11, 12].

Large diversity deals with a set of customer's functional requirements. Each one can be derived in a series of increasing service levels. Consequently, the functional diversity results from the combinatorial gathering of the service levels of the customer's requirement.

Conversely, the admissible decompositions of the product family can be defined as a set of admissible sub-assemblies, each one derived in a number of variants. Each variant of a sub-assembly has been designed in order to meet a given service level of a customer's requirement.

The goal is to define a family of products P = {(ref_sub-assembly, ref_variant)}, that can match any demand D={(customer_requirement, service_level)} while controlling the supply chain performance.

As it is assumed that the service levels are defined according to an order relation, meaning that a variant corresponding with a given service level can fulfill all the requirements of lower service levels, two extreme strategies can be identified in order to define the family of products P:

1. a single pack approach, where a single product gathers the highest variant of each sub-assembly and therefore matches all the service levels of all the customer requirements. This extreme standardization enables short customer lead times and savings in the operating costs (scale economies, safety stocks, dedicated process), but with great over-equipment costs.
2. a tailored or modular approach, where a product is customized for each customer demand. In order to control the operating costs and customer lead-time, this approach needs a modular decomposition of the family and highly flexible processing means.

A good solution would be a compromise between these two strategies.

In the pack strategy, the set of customer requirements will be decomposed into a finite hierarchy of demand segments for which specific packs will be designed. This defines a multi-pack approach. But, it would be interesting to design some sub-assemblies that can be common to a subset of packs. This means that some modules will appear in the decomposition of the packs.

In the modular approach, a specific module can hardly be designed per service level of the customer requirements. Thus, a pack strategy is usually developed in order to select the module variants.

Consequently, pack and modular approaches should be combined in order to define a good product family. For example, household electrical appliance families are usually decomposed in packs (per market range and per trademark), computer families are much more modular, while automotive families result from a compromise between a pack and a modular approach: a car is usually defined according to a market range level (bottom, middle, up-of-the range but also family, sport, touring) plus some optional requirements (color, road map computer, loudspeakers...).

Previous works dealing with the "Design for Supply Chain" concepts have shown various interests in adapting the product architecture and decomposition in order to enhance the costs and lead-time of a given supply chain. Authors mainly quantify the safety stocks cutting down due to the diversity decrease through the application of various principles: time, place or form postponement of a product variant [13,6], operation reversals resulting in component reversal in the bills-of-material [14], product and process modularization or standardization [4]. More recently, Van Hoek [21] synthesized the developments of postponement applied to a supply chain. Anderson [1] and Pine II [19] underlined the integration of these concepts in order to propose mass-customizable products. Our purpose is to integrate these product architectural and the supply chain design decisions, that results in a global design for supply chain process.

Two interactive processes are proposed (figure 13-1).

– The first process, the bills-of-material provider process, should be able to define easily and quickly error free bills-of-material for a set of pre-identified customer requirements, demand segments or packs. It is a pure product design process. It is only considered at the preliminary design step that deals with the definition of some principles of design and of the product architecture [17].
– The second process would optimize the compromise "over equipment cost / reference management cost", through the simultaneous definition of (i) the supply chain (where to manufacture, to assemble and to store) and (ii) the selection of a decomposition of the product family.

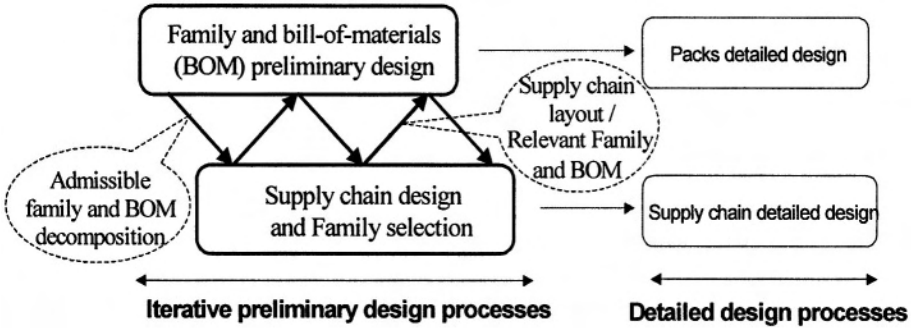

Family and bill-of-materials (BOM) preliminary design
Packs detailed design
Supply chain layout / Relevant Family and BOM
Admissible family and BOM decomposition
Supply chain design and Family selection
Supply chain detailed design
Iterative preliminary design processes
Detailed design processes

Figure 13-1. Interactive design process of the product family and supply chain

The interactivity between these two processes comes from the "try and evaluate" or iterative design approach that can be handled as illustrated in Figure 13-1. For example, the first process can be achieved for some packs then the second process provides a first result, then other packs or modules can be proposed and optimized in order to improve compromise of cost in an iterative way.

The bills-of-material provider process is not in the focus here. Details referring to an approach that supports this process using a constraint-based generic model that gathers product functional and physical knowledge can be found in [9, 10].

Among the works dealing with supply chain design, strategic planning is interested in optimizing the layout of a complete supply chain [2, 7, 5]. Basically, the strategic design of a supply chain select facilities to be opened, fixes their capacity, defines the shipping channels to use and quantifies the product flows in order to minimize the sum of the supply chain total costs. This problem is close to the one faced in the second design process.

Many models have been formulated for this problem, called GSCM (Global Supply Chain Model) [7]. All these models consider the product bill of material (BOM) as a hierarchical tree of physical items with only "AND" nodes. Therefore, given a specific set of finished products with relevant bills-of-material and an extended demand volume per product, these GSCM optimize, thanks to a Mixed Integer Linear Program, the relevant supply chain layout.

However, the bills-of-material provider process should result in a series of admissible product decompositions. In order to optimize the product family, we propose to gather the set of all possible decompositions in a single generic hierarchical tree and call it G-BOM for "Generic Bill-Of-Material", as shown in part 2. Part 3 depicts a mixed integer linear model that optimizes the total costs of a supply chain while selecting a product family decomposition from the G-BOM. An example is detailed in part 4.

2. DESCRIPTION OF THE ADMISSIBLE FAMILY DECOMPOSITIONS

During a planning process, the bill-of-material is used in order to compute net requirement of items given the demand of final products. Here, a Generic Bill-Of-Material (G-BOM) expresses the admissible ways to distribute the customer demand, and thus net requirements, to the designed elements.

The G-BOM remains a directed graph without cycles, in which nodes refer to items and arc values refer to the necessary quantity of child item per parent item. To match the extension, the key idea is to add new notions of "logical item" versus "physical item" and "exclusive OR" bill-of-material nodes versus "AND" nodes (figure 13-2):

1. An "exclusive OR" node is introduced to show that one and only one item must be selected among all the child items of the node. It allows representing the choice of existence of a child item. An "AND" node expresses that all its child items must be gathered in order to make it.

2. A logical item is only necessary for the G-BOM expression. Due to its logical meaning, it can neither be manufactured, nor stored, nor shipped. During a planning process, it allows to compute a required quantity resulting from its parent items, or from customer demands, that must be fulfilled by child items. Conversely, a physical item can be stored, manufactured or shipped. Logical items are usually used in order to formalize the customer demand or the result of choices within the G-BOM.

In the case of Figure 13-2, the G-BOM expresses the potential existence of allowed combination of packs *(P1, P2, P3, P4),* plus the choice of a sub-assembly per pack. The G-BOM can be read as follows:

- The demand is split up into four market segments with a given percentage of market (respectively 20%, 30%, 30%, 20%).
- Each market segment can be fulfilled either by a specifically dedicated pack, or by the solution of the next higher demand segment. For example, Segment_1 can be fulfilled either by a pack solution P1, or by the solution retained for Segment_2.
- Several design solutions have been developed for a same pack, so, for a same set of customer requirements. For example, there are three admissible design solutions for pack P2 (BOM_2.1 to BOM_2.3).

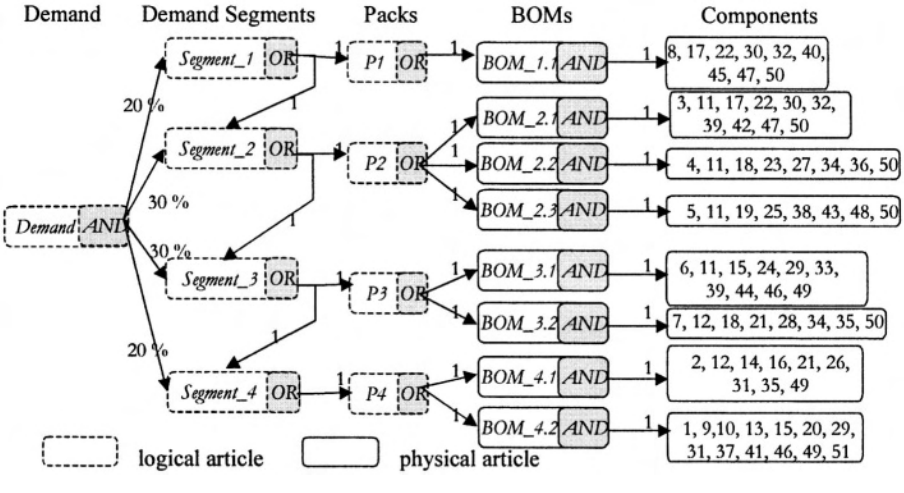


Figure 13-2. Example of a Generic Bill-Of-Material of a multi-pack approach

Consequently, the eight combinations of packs allowed are: *{P4}, {P1, P4}, {P2, P4}, {P3, P4}, {P1, P2, P4}, {P1, P3, P4}, {P2, P3, P4}* and *{P1, P2, P3, P4}.* But, 48 different ways to affect the demand to the component items result from considering the admissible pack decompositions. "Demand", "Demand-Segment" and "Pack" items are logical items that cannot be delivered to a customer, but enable to express how the customer demand results on the physical items (“BOM” and “Components”). Some “component” items (“11” for example), appear within several “BOM” items (“BOM1.1” and “BOM2.1”). They are modules on which scale economies can be made if all the parent BOM items are selected.

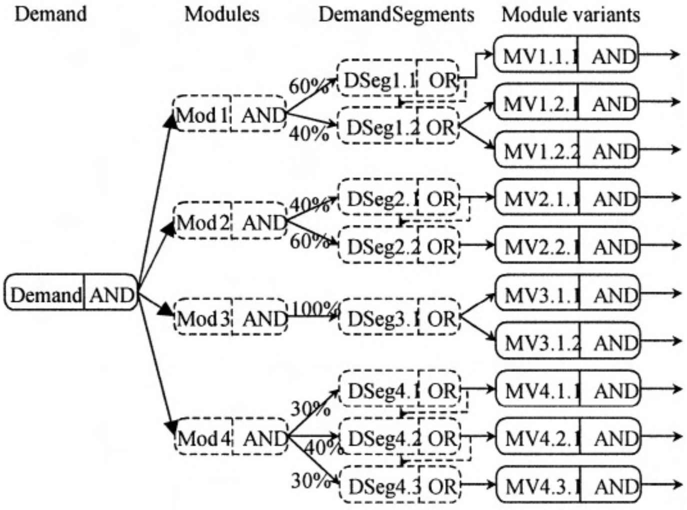


Figure 13-3. Example of a Generic Bill-Of-Material of a modular approach

In the example of Figure 13-3, the G-BOM results from a modular design approach: the customer requirements are split up into four “Modules”. Each module is also split up into “Market Segments” and “Module Variants” through a pack approach. “Module Variants” (MV1.1.1 to MV4.3.1) are physical items. Their assembly is modeled through the gathering of the modules, so through the relation between the “Demand” item and the “Module” items. Consequently, “Demand” is a physical item while “Modules” and “Demand-segments” items are logical.

3. COST OPTIMISATION THROUGH FAMILY AND SUPPLY CHAIN DESIGN

Basically, the global supply chain design models [2, 5, 7] are defined with:

– continuous variables: one per item, time period, facility with three kinds of activities (manufacturing, inventory and shipping),
– binary and integer variables: one per facility describing whether the facility is used or not, and the number of resource lines opened,
– classical strategic planning linear constraints between theses variables,

– a cost function to minimize, that gathers variable costs (manufacturing, inventory and shipping) and fixed costs (item references, facilities and resource lines fixed costs).

Many authors have introduced extensions of the GSCM in order to consider specific problems such as: the effect of taxes, duties, exchange prices that force to optimize the global total profit instead of the total costs [2, 3]; scale economies on manufacturing or inventory systems [16]; the choice of technological manufacturing systems [18]; the effect of uncertainties on demand or on exchange rates [15, 8].

Here, a classical GSCM is extended to consider the G-BOM. The resolution will provide (i) the product family retained, (ii) the list of opened facilities and resource lines and (iii) the optimal supply chain operating cost.

Therefore, let us add, for representing the Generic Bill-Of-Material:

– binary variables: one for each item of the G-BOM describing whether the item exists or not, one for each G-BOM link **parent(OR)→ child** (arising only with the "OR" nodes) describing if the parent item chooses to require the child item,

– constraints between these binary variables.

A mixed integer linear programming model is defined in the following as an extension of basic models such as: (i) a single shipping channel is available between any two facilities; (ii) manufacturing or shipping activities have much smaller lead-times than the time period, and thus are supposed continuous.

Literally, the model can be expressed as:

Minimize total cost = Fixed cost relevant to item existence, facility, resource lines and shipping channel existence (1) + Variable manufacturing, inventory and shipping costs (2).

Subject to constraints:

– Generic Bill-Of-Material constraints (constraints 3 to 9)

– Item flow conservation constraints (constraints 10 to 13)

– Shipping constraints (constraints 14, 15)

– Inventory constraints (constraints 16, 17)

– Capacity constraints (constraints 18)

– Binary variables domain constraints (constraints 19)

But before detailing the model, let us depict the notations and variables.

3.1 Notations of sets, costs and parameters

The various sets used in the model:

T is the set of time periods,

C is the set of customers,

U is the set of production/inventory facilities,

R is the set of types of resource lines,

P_r is the set of items manufactured on the type of line *r*,

$P = \Phi \cup \overline{\Phi}$ is the set of items (physical Φ /logical $\overline{\Phi}$),

$P_c \subset P$ is the sub-set of items with external demand relevant to customer *c*,

BOM_p is the set of child items of an item *p* (with "AND" or "OR" node),

BOM_p^{-1} is the set of parent items of an item *p*,

$P^\wedge$ (resp. $P^\vee$) is the set of items with "AND" node (resp. with node "OR").

The fixed and variables costs of items, facilities and shipping channels:

ECF_p is the fixed cost of existence of a physical item p,

$UCFO_u$ is the fixed cost of opening facility $u \in U$,

$UCFO_{ru}$ is the fixed cost of opening a line *r* at facility $u \in U$,

MCF_{uv} is the fixed cost of a shipping channel from facility $u \in U$ to $v \in U \cup C$,

UCV_{pu} (resp. SCV_{pu}) is the variable cost to make (resp. store) an item *p* at facility $u \in U$,

MCV_{puv} is the variable cost of shipping one item *p* between facilities $u \in U$ and $v \in U \cup C$.

Parameters:

D is the duration of a time period,

M_∞ is the maximum number of items to manufacture, to store or to ship on a time period,

Dem_{pct} is the demand from customer facility *c* for item *p* during period *t*,

α_{pq} is the units of child item *p* required to make one unit of parent item *q*,

$MaxLP_{ru}$ is the maximum number of resource lines of type *r* at facility *u*,

$URes_{rp}$ is the time of a resource line of type *r* necessary per product $p \in P_r$,

Cf_{pu} (resp. Ct_{puv}) is the manufacturing (resp. shipping) lead-time of product *p* at facility $u \in U$ (resp. up to facility $v \in U \cup C$) defined as a percentage of *D*,

Cov_{pu} is the percentage of manufacturing and shipping lead-time of product *p* to get at facility $u \in U$ as safety stock.

3.2 Binary/continuous decision variables

Binary variables:

$\lambda_p = 1$ if item *p* exists, otherwise 0,

X_u = 1 if facility u is opened, otherwise 0,

λ_{pq} = 1 if item $q \in P^{\vee}$ requires item p as a child, otherwise 0. This binary variable selects a bill-of-material link for item q with an "OR" node.

Z_{uv} = 1 if the shipping channel between facilities u and v exists, otherwise 0,

$LPO_{ru} \in \mathrm{N}$ is the integer number of resource lines of type r opened at facility u.

Continuous variables:

X_{put} is the net requirement of the item p on facility u at period t. For a physical item, this net requirement is also the quantity manufactured,

X_{pqut} is the net requirement associated to a parent item q with node "OR" using the child item p at facility u at period t, if the link $q \rightarrow p$ exists ($\lambda_{pq} = 1$),

Y_{put} is the amount of a physical item p stored at facility u at the end of period t,

Z_{puvt} is the amount of item p shipped from facility u to facility v during period t,

From these notations and decision variables, the mathematical programming model can be formulated.

3.3 Mathematical formulation

3.3.1 The objective

The model minimizes the sum of total costs: fixed cost relevant to the items (1.1), plus facility, resource lines, shipping channel existences (1.2) plus variable manufacturing, inventory and shipping costs (2).

$$\text{Total cost} = \sum_{p}^{\Phi} ECF_p.\lambda_p \tag{1.1}$$

$$+\sum_{u}^{U} UCFO_u.X_u + \sum_{u}^{U} \sum_{v}^{U \cup C} MCF_{uv}.Z_{uv} + \sum_{r}^{R} \sum_{u}^{U} UCFO_{ru}.LPO_{ru} \tag{1.2}$$

$$+\sum_{p}^{\Phi} \sum_{u}^{U} \sum_{t}^{T} \left[UCV_{pu}.X_{put} + SCV_{pu}.Y_{put} + \sum_{v \neq u}^{U \cup C} MCV_{puv}.Z_{puvt} \right]. \tag{2}$$

3.3.2 Generic Bill-Of-Material constraints

The G-BOM constraints express the existence of items and links of the G-BOM. According to the type of the G-BOM node ("AND" or "OR"), different constraints restrict the binary variables λ_p. Recall that variables λ_{pq} are only defined if item q corresponds to an "OR" node in the G-BOM.

For any kind of node

A net requirement is associated to an item p on facility u if and only if this item exists and the relevant facility is opened (3):

$$X_{put} \le M_\infty \lambda_p \text{ and } X_{put} \le M_\infty X_u \qquad \forall p \in P, u \in U \cup C, t \in T\,. \tag{3}$$

Constraint (4) expresses that when an item p (without external demand) is selected, either one of its parent items with an "AND" node exists, or a link relevant to an "OR" node exists.

$$\lambda_p \le \sum_q^{BOM_p^{-1} \cap P^\wedge} \lambda_q + \sum_q^{BOM_p^{-1} \cap P^\vee} \lambda_{pq}\,, \qquad \forall p \in P - P_c\,. \tag{4}$$

For "OR" nodes

Constraints (5, 6, 7, 8) are only relevant for items q with an "OR" node ($q \in P^\vee$). Constraint (5) ensures that a link relevant to an "OR" node exists if and only if both parent and child items exist. Constraint (6) stands that if an item $q \in P^\vee$ exists, one and only one link must be selected between q and its child items. Constraint (7) means that the existence of a net requirement of an item q triggers a gross requirement on its child items. Constraint (8) makes sure that if a link $q \rightarrow p$ exists, the net requirement associated to the item q requires the child item p.

$$\lambda_{pq} \le \lambda_q \text{ and } \lambda_{pq} \le \lambda_p \qquad \forall q \in P^\vee, p \in BOM_q \tag{5}$$

$$\sum_p^{BOM_q} \lambda_{pq} = \lambda_q \qquad \forall q \in P^\vee \tag{6}$$

$$\sum_p^{BOM_q} X_{pqut} = X_{qut}\,, \qquad \forall q \in P^\vee, u \in U, t \in T\,, \tag{7}$$

$$X_{pqut} \le M_\infty \lambda_{pq}\,, \qquad \forall q \in P^\vee, p \in BOM_q, u \in U, t \in T\,. \tag{8}$$

Constraints (5, 6, 7, 8), altogether, imply that:

$$X_{pqut} = X_{qut} \Leftrightarrow \lambda_{pq} = 1\,,\ \forall q \in P^{\vee}, p \in BOM_q, u \in U, t \in T\,.$$

For "AND" nodes

Constraint (9) ensures that the existence of a parent item $q \in P^{\wedge}$ implies the existence of all its child items.

$$\lambda_p \geq \lambda_q\,, \qquad \forall q \in P^{\wedge}, p \in BOM_q \qquad (9)$$

3.3.3 Flow conservation constraints

Flow conservation constraints specify that for each facility *u* and during each time period *t,* the inventory variation of an item *p* must be equal to the sum of quantities generated in the facility *(X)* and coming from other facilities (*Z*) minus the sum of quantities shipped to other facilities (*Z* and *Dem)* and minus the sum of quantities consumed in the facility to satisfy the gross requirements associated to the parent items *(X).*

According to our G-BOM definition: (i) a logical item can neither be manufactured, stored nor shipped, but it can generate a gross requirement in any facility, (ii) a physical item can neither be manufactured nor stored in a customer facility, but it can be shipped to it.

Therefore, according to the type of facility (manufacturing or customer), and the type of item (physical or logical), the flow conservation changes (see Figure 13-4).

So, flow constraints are formulated for each case:

– Constraints (10): physical items in production facility,

$$\forall p \in \Phi, u \in U, t \in T, \qquad Y_{put} - Y_{put-1} =$$

$$= X_{put} + \sum_{v \neq u}^{U} Z_{pvut} - \sum_{v \neq u}^{U \cup C} Z_{puvt} - \sum_{q}^{P^{\vee} \cap BOM_p^{-1}} \alpha_{pq} X_{pqut} - \sum_{q}^{P^{\wedge} \cap BOM_p^{-1}} \alpha_{pq} X_{qut} \qquad (10)$$

– Constraints (11): logical items in production facility,

$$\forall p \in \overline{\Phi}, u \in U, t \in T\,,$$

$$0 = X_{put} - \sum_{q}^{\left(P^{\vee} \cap BOM_p^{-1}\right)} \alpha_{pq} X_{pqut} - \sum_{q}^{\left(P^{\wedge} \cap BOM_p^{-1}\right)} \alpha_{pq} X_{qut}\,. \qquad (11)$$

– Constraints (12): physical items in customer facility,

$$\forall p \in \Phi, c \in C, t \in T\,,$$

$$Dem_{pct} = \sum_{v}^{U} Z_{pvct} - \sum_{q}^{P^{\vee} \cap BOM_p^{-1}} \alpha_{pq} X_{pqct} - \sum_{q}^{P^{\wedge} \cap BOM_p^{-1}} \alpha_{pq} X_{qct}\,. \qquad (12)$$

– Constraints (13): logical items in a customer facility,

$$\forall p \in \overline{\Phi}, c \in C, t \in T\,,$$

$$Dem_{pct} = X_{pct} - \sum_{q}^{P^{\vee} \cap BOM_p^{-1}} \alpha_{pq} X_{pqct} - \sum_{q}^{P^{\wedge} \cap BOM_p^{-1}} \alpha_{pq} X_{qct}\,. \qquad (13)$$

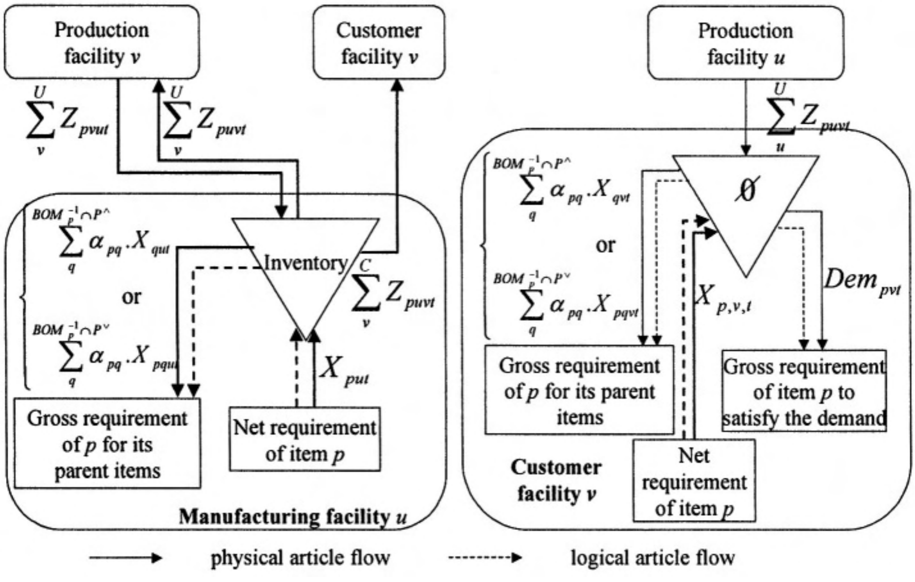

Production facility v
Customer facility v
Production facility u
$\sum_{v}^{U} Z_{pvut}$
$\sum_{v}^{U} Z_{puvt}$
$\sum_{u}^{U} Z_{puvt}$
$\sum_{q}^{BOM_p^{-1} \cap P^{\wedge}} \alpha_{pq} . X_{qut}$
or
$\sum_{q}^{BOM_p^{-1} \cap P^{\vee}} \alpha_{pq} . X_{pqut}$
Inventory
$\sum_{v}^{C} Z_{puvt}$
X_{put}
Gross requirement of p for its parent items
Net requirement of item p
Manufacturing facility u
$\sum_{q}^{BOM_p^{-1} \cap P^{\wedge}} \alpha_{pq} . X_{qvt}$
or
$\sum_{q}^{BOM_p^{-1} \cap P^{\vee}} \alpha_{pq} . X_{pqvt}$
∅
$X_{p,v,t}$
Dem_{pvt}
Gross requirement of p for its parent items
Gross requirement of item p to satisfy the demand
Customer facility v
Net requirement of item p
physical article flow
logical article flow

Figure 13-4. Physical/logical item flows through a manufacturing or a customer facility

3.3.4 Shipping constraints

Constraints (14) and (15) mean that shipping quantities depend on the existence of facilities, shipping channels and items.

$$Z_{uv} \le X_u \text{ and } Z_{uv} \le X_v\,, \qquad \forall u \in U, v \in U - \{u\} \cup C\,, \qquad (14)$$

$$\forall p \in \Phi, u \in U, v \in U - \{u\} \cup C, t \in T,$$
$$0 \le Z_{puvt} \le M_{\infty} Z_{uv} \text{ and } Z_{puvt} \le M_{\infty} \lambda_p. \tag{15}$$

3.3.5 Inventory constraints

Constraint (16) shows that the inventory depends on existence of items and facilities.

$$Y_{put} \le M_{\infty} X_u \text{ and } Y_{put} \le M_{\infty} \lambda_p, \qquad \forall p \in \Phi, u \in U, t \in T. \tag{16}$$

A key point of supply chain design is to evaluate whether the postponement of an item can be useful. Such a choice has an impact on the item’s lead-time. Items with long lead-times require higher safety stocks.

As a consequence, we model the safety stock of an item in a facility as a percentage of the flows during the item’s lead-time. The flows under question will be the item internal production flow, the item supplying flow from other facilities, and the item delivering flow to customers.

Thus: $\forall p \in \Phi, u \in U, t \in T$,

$$Y_{put} \ge Cov_{pu} \left(Cf_{pu} X_{put} + \sum_{v \ne u}^{U} Ct_{pvu} Z_{pvut} + \sum_{c}^{C} Ct_{puc} Z_{puct} \right). \tag{17}$$

3.3.6 Capacity constraints

These constraints express that enough resource lines must be opened into opened facilities in order to manufacture the desired quantities.

$$\sum_{p}^{P_r} X_{put} U \operatorname{Re} s_{rp} \le D \cdot LPO_{ru} \le D \cdot MaxLP_{ru} X_u, \quad \forall r \in R, u \in U, t \in T. \tag{18}$$

3.3.7 Binary variable constraints

$$X_u, Z_{uv}, \lambda_p \in \{0,1\}, \ LPO_{ru} \in \mathrm{N}, \qquad \forall u \in U, v \in U \cup C, \forall p \in P$$
$$\text{and } \lambda_{pq} \in \{0,1\}, \qquad \forall q \in P^{\vee}, p \in BOM_q \tag{19}$$

4. EXPERIMENTAL EVALUATION

The previous model has been coded in C++ with Ilog-Cplex 6.5 library. Some experimentations were conducted on the problem of a supplier of a car manufacturer.

The Generic Bill-Of-Material is the one of Figure 13-3 and depicts 48 specific bills-of-material. Products are assembled from basic components (components 1 up to 48) and from computers (components 49, 50 and 51).

There are 8 manufacturing facilities (some are specialized for computer manufacturing) and 4 customer ones distributed over Europe (figure 13-5). One major constraint is that final products must be shipped through a synchronous delivering from so called "synchronous" facilities that must be close to the customer. But basic components can be manufactured and assembled with the computers either at a "synchronous" facility or at a component one.

With a demand expressed on 8 periods, the linear problem gathers 30849 constraints, 20073 real variables and 559 integer variables. It has been solved with a SUN UltraSPARC/143 MHz/192Mo-RAM in 744 seconds.

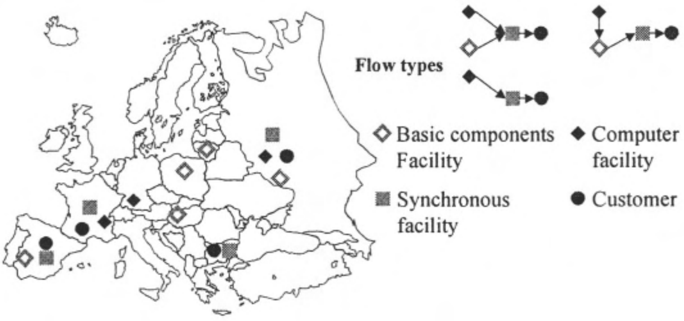


Figure 13-5. The supply chain that must be designed

To show the interest of our approach, the same calculation has been processed with some simplified generic bills-of-material, meaning that each Generic Bill-Of-Material represents a smaller combinatory. This is achieved by fixing choices on some of the "OR" BOM nodes while keeping the same optimal solution. Computation time decreases with the combinatory, as shown in Figure 13-6, until the duration reaches 115 seconds when combinatory equals 1. For this last case, the Generic Bill-Of-Material corresponds with the bill-of-material of the optimal solution. Thus, it is better to solve the problem once (744 seconds) than solving 48 problems (115 seconds each).

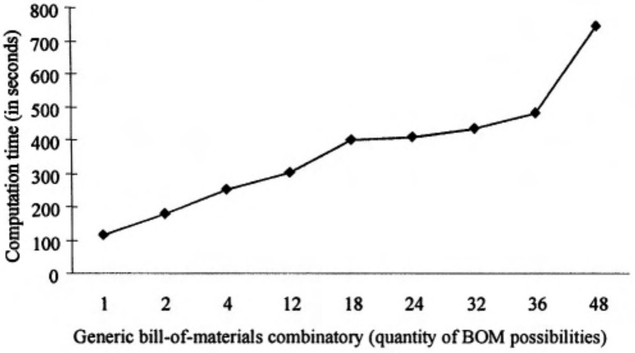


Figure 13-6. Computational time versus the combinatory of the Generic BOM

5. CONCLUSION

The approach developed in this chapter allows to define a product family that matches a demand presenting a large diversity and the relevant supply chain while minimizing the cost compromise "over equipment cost / references management cost".

The approach and relevant models fit the first steps of the design of a new product family when supply chain aspects are important and must be taken into account. This issue is very common for example in automotive industry, electrical-appliance or PC industry. This result can be considered as a step towards the integration of the product and supply chain designs in a context of concurrent engineering.

REFERENCES

1. Anderson D.M., 1998, *Agile product developmentfor mass customisation,* Mc Graw Hill, ISBN 0-7863-1175-4.
2. Arntzen B.C., Brown G.G., Harrison T.P., Trafton L.L., 1995, Global supply chain management at digital equipment corporation, *Interfaces,* 25 (1), 69-93.
3. Cohen M.A., Lee H.L., 1989, Resource deployment analysis of global manufacturing and distribution networks, *Journal ofManufacturing and Operations Management,* 2, 81-104.
4. Erixon G., 1996, Design for Modularity, In *Design for X: concurrent engineering imperatives,* Huang G.Q. Edts, Chapman&Hall, 356-379.
5. Ganeshan R., Jack E., Magazine M.J., Stephens P., 1999, A taxonomic review of supply chain management research, *Quantitative Models for SCM,* Kluwer Academic, Chap. 27, 839-879, Boston.
6. Garg A., Tang C.S., 1997, On postponement strategies for product families with multiple points of differentiation, *IIE Transactions,* 29, 641-650.

7. Goetschalckx M., 2000, Strategic Network Planning, in *Supply chain management and advancedplanning,* H. Stadler & C. Kilger ed., Springer, Heidelberg, Germany.
8. Goetschalckx M., Ahmed S., Shapiro A., Santoso T., 2001, Designing flexible and robust supply chains, *IEPM Conference,* Quebec City, Canada.
9. Hadj-Hamou K., Aldanondo M., Lamothe J., 2003, Product with large diversity: an approach towards simultaneous design of product and supply chain, *International Conference on Engineering Design ICED'03,* Stockholm, August 19-21.
10. Hadj-Hamou K., Caillaud E., Lamothe J., Aldanondo M., 2001, Knowledge for product configuration, *Int. Conf. on Engineering Design,* Glasgow, Scotland.
11. Lee H.L., 1993, Design for supply chain management: concept and examples, *Perspectives in operations Management,* Sarin R. Edts, Kluwer academic Publishers, Boston, MA, 835-847.
12. Lee H.L., 1995, Product universality and design for supply chain management, *Production Planning and Control,* 6 (3), 270-277.
13. Lee H.L., Billington C., 1994, Designing products and processes for postponement, *Management of design: engineering and management perspectives,* Dasu S. and Eastman C. Edts, Kluwer Academic Publishers, Boston, 105-122.
14. Lee H.L., Tang C.S., 1998, Variability reduction through operations reversal, *Management Science,* 44 (2), 162-172.
15. Lucas C.A., Mitra G., Mirhassani S., 1999, Supply chain planning under uncertainty, *Quick Response in the Supply Chain,* Edited by E. Hadjiconsrantinou, Springer.
16. Mattel A., Vankatadri U., Optimizing supply network structures under economies of scale, *Int. Conf. on Industrial Engineering & Production Management,* Glasgow, 1999.
17. Pahl G., Beitz W., 1996, *Engineering Design: a Systematic Approach,* Springer-Verlag, London, 2nd edition.
18. Paquet M., Martel A., Desaulniers G., 2001, Including Technology selection decisions in manufacturing network design models, *International Conference on Industrial Engineering and Production Management IEPM 2001,* Quebec City, Canada.
19. Pine II B. J., 1993, *Mass customization: the newfrontier in business competition,* Harvard Business School Press Ed., Boston.
20. Prasad B., 1996, *Concurrent Engineering Fundamentals: Integrated Product and Process Organization,* 1, Prentice Hall.
21. Van Hoek R. I., 2001, The rediscovery of postponement: a litterature review and directions for research, *Journal of Operations Management,* 19, 161-184.

Chapter 14

SALES AND OPERATIONS PLANNING OPTIMISATION

Contribution and limits of linear programming

Patrick Genin, Samir Lamouri, André Thomas

Abstract: Operations' planning requires strategic decisions on inventories levels, on demands and operations constraints. The importance of these decisions leads to elaborate and optimise Sales and Operations Plans on a planning time fence at least as long as the budget. Models using linear programming give the "optimal" strategy but do not resist frequent changes in parameters. Other mathematical tools as well as Taguchi methods are approaches to achieve a simple but robust compromise.

Key words: sales and operations planning, optimisation, linear programming, robustness, planning.

1. INTRODUCTION

Today, Supply Chain Management defines the global level of resources for each activity within the firm (i.e. production, maintenance, etc.) in order to satisfy the actual sales forecasts. Planning production allows to make on-time arrangements to satisfy sales with needed quantities and due dates at the smallest cost. These three objectives cannot be simultaneously achieved. The planning decision is the result of a balance between on-time deliveries, risks on inventories and operations costs.

The process "Sales and Operations Plan" (S&OP) builds the best strategy with time fence at least as long as the budget to realise this balance for product groups [13]. The expected performance for other activities is deduced on a mid/long term.

The traditional S&OP calculation is based on graphical techniques or on Linear Programming models [3]. This chapter sets out how an approach of S&OP by linear programming, can balance inventories, on-time deliveries and operations costs but also points out the limits in robustness of the strategy.

2. SALES AND OPERATION PLAN (S&OP)

S&OP puts into effect strategic objectives established by management when dealing with the strategic plan. It is the link between the sales planning and operations. S&OP is entirely integrated with information and the demand management systems. It drives the execution of the different Master Planning Schedules (MPS). S&OP is a useful tool for prospective analysis within medium to long-term range.

As the operations system is not flexible enough to follow sales day-to-sales' variations, adjustments are needed within the planning level. Sales are uncertain data with quick and unpredictable variations. If the demand could be exactly forecasted, the workload on resources should react the same way. However this is not always possible. The number of machines is fixed, training new staff takes time and the negotiations with suppliers have an impact on lead-time and quantities produced. The firm has to answer the following question: How can the production system capacity keep up with fluctuations in the sales volumes? It is the key role of S&OP to answer that question [8].

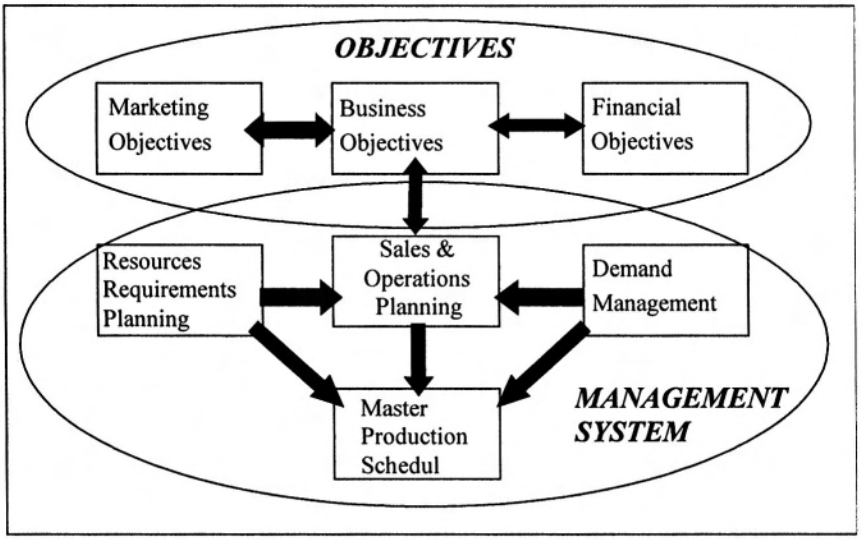


Figure 14-1. S&OP and industrial management functions

The S&OP anticipates the sales' evolutions in products families in order to adapt the operations and the supply chain system to its market. At that level, budgetary capacity is going to be taken into account. The S&OP will financial capacity, inventories, workforces and rough-cut capacities availability to turn the sales and strategic objectives (market shares...) into activities to complete on the mid term. The different sub-systems are linked together (Figure 14-1).

The planning horizon will often be 18 months long revue on a monthly basis. The parameters are set to take account special events such as promotions, special agreements, ...

Several simulations could be performed in order to determine the optimal strategy that will minimise the total cost while maximising the sales.

3. S&OP OPTIMISATION TECHNIQUES

Two approaches are frequently used: graphical methods and Linear Programming optimisation by [4, 5, 6].

The spreadsheets and graphical methods are widespread because of the simplicity of use and understanding. The plans are obtained with few variables settled at a time to let the manager compare the forecasted demand to the existing capacity. These graphical methods work by iterations; they identify different integrated and feasible plans where cost is not necessarily the lowest possible. The manager must consequently use his feelings to determine the appropriate plan.

Graphical methods generally proceed in 5 steps as follow:

1. Evaluation of monthly demand;
2. Evaluation of monthly capacity with standard working time, overtime or subtracting;
3. Identification of labour costs, overhead cost, etc.;
4. Strategic evaluation of changing workforce or inventory level;
5. Setting-up alternatives and balancing total costs.

These management tools help to evaluate different strategies but do not generate them. Whereas decision makers expect a systematic approach that considers the whole costs and gives an efficient answer to that problem. Mathematical models, using linear programming propose such an approach [1]. In the following, an application in an industrial context is described.

4. AN EXAMPLE OF A COMPANY

Vallourec Précision Etirage (VPE) produces steal tubes in parts or full length for automotive markets (layer 1, 2 and 3 supplier) and for mechanical markets (heaters, boilers, circuits). The Supply Chain initiative reengineers business process (industrial and administrative) in order to reach 98 % of on-time delivery.

Settled in 4 production entities, The 10 flow-shops ensure the production of the 70 commercial families by working five days a week three shifts per day. The demand by products family “the load” is different each month. On the contrary the capacity is relatively stable.

The S&OP is the monthly process for updating the tactical planning by consolidating production and demand on a 12 months time-fence. The steps are (Figure 14-2) [2]:

1. Demand Forecast calculation in the sales department in term of commercial families and production lines;
2. Load calculation and capacity balancing for each production line by the line manager as well as calculation by the supply chain manager at the firm level;
3. Creation of scenarios and actions plans for each line and for the whole company;
4. Consolidation of resources requirements and availabilities and action plans validation by the supply chain manager;
5. Monthly meeting to present the scenarios and choice of the strategy by the steering comity.

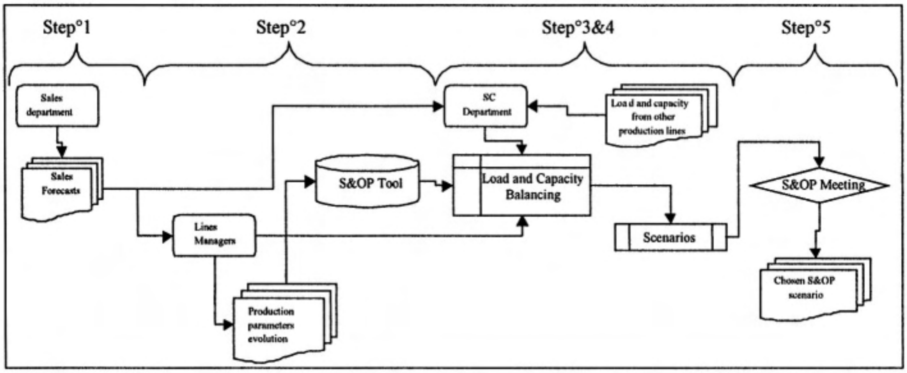


Figure 14-2. S&OP Process of VPE

The firm adjusts a set of logistic variables to spread the workload during its S&OP process:

- Seasonal inventories;
- Capacity adjustments (working during weekends and bank holidays);
- Subcontracting (limited for strategic reasons);
- Backorders or inventories;
- Priorities by products family on production resources.

The determination of scenarios and associated total costs is difficult to make by hands. Linear Programming makes it possible to find the optimum for a whole set of given conditions.

5. THE LINEAR PROGRAMMING MODEL PROPOSED FOR VPE

The model presented is simplified: we do only consider one production line and only one products family. Moreover, data are truncated for confidentiality reasons.

The S&OP determines for the considered products family:

- production level;
- inventory level;
- subcontracting level;
- number of additional working days or non-working days;
- S&OP over-cost engaged by the scenario.

The main over-cost is due to production. It is composed of the over-costs caused by additional or non-working days and of subcontracting. In addition to this production over-cost, VPE considers the carrying costs. The Business Plan aims at a service rate of 100 %, that is to say no delay, and a level of stock lower than 3 days, 195 T Consequently, algebraic stock cannot be negative. It thus lies between 0 and 195 T. That last constraint does not consider the seasonal inventory.

Table 14-1. S&OP data

Item	Value
Unit inventory cost by period, c^I	190 €
Unit backlog cost by period, c^B	2 300 €
Unit cost incurred per additional day, c^{OV}	800 €
Unit cost incurred by non-production day, c^{NP}	1 300 €
Unit cost for subcontracting, c^{SC}	600 €
Beginning inventory, I_0	100 T
Available capacity expressed in unit per day, e	65 T/d

5.1 The variables definition

t is the period index. T is the horizon length, 12 in our case. D_t represents the forecasted demand for period t. N_t corresponds to the standard working days in period t, N_t^* is the maximum of working days per period t. u_t is the standard capacity in period t. It is determined by the formula (1). O_t^* is the maximum overtime capacity in period t. Relation (2) gives it. The values used are presented in Table 14-1.

$$\forall t,\ u_t = e \times N_t, \tag{1}$$
$$\forall t,\ O_t^* = e \times [N_t^* - N_t]. \tag{2}$$

5.1.1 The decision variables

O_t is the number of tons manufactured in additional days within period t. S_t is the number of tons manufactured in subcontracting during period t. S_t^* is the upper limit to subcontracting. N_t is the number of tons which have not been produced during the non-working days in period t. I_t is the inventory level at the end of period t. B_t is the backlog level at the end of period t. P_t is the total production carried out.

5.1.2 The objective function

The objective of the problem is represented by the minimization of the sum of the different cost factors, i.e. the costs for production in overtime and for non-production, subcontracting, inventory, backlogs. It determines the over-cost of the determined scenario (3):

$$\sum_{t=1}^{T} c^{OV} \times O_t + c^{NP} \times N_t + c^{SC} \times S_t + c^{I} \times I_t + c^{B} \times B_t\ . \tag{3}$$

5.1.3 The constraints

The following describe the constraints in the model.

$$\forall t,\ P_t = u_t + O_t - N_t + S_t, \tag{4}$$
$$\forall t,\ P_t = D_t + I_t - I_{t-1} + B_t - B_{t-1}, \tag{5}$$
$$\forall t,\ I_t \le I_t^*, \tag{6}$$
$$\forall t,\ S_t \le S_t^*, \tag{7}$$
$$\forall t,\ O_t \le O_t^*, \tag{8}$$
$$\forall t,\ N_t \le u_t, \tag{9}$$
$$\forall t,\ 0 \le O_t, N_t, S_t, I_t, B_t\ . \tag{10}$$

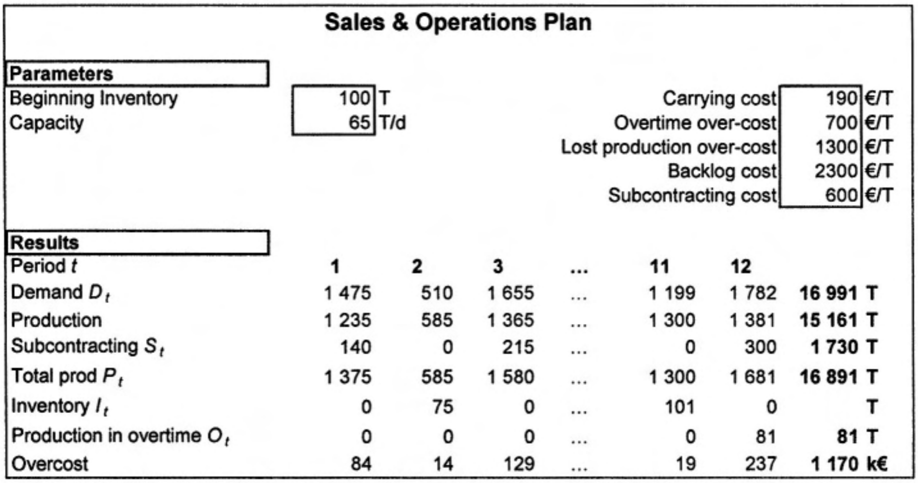
Constraints (4) state that in each period the whole production is obtained with the standard capacity plus or minus what is produced in overtime or not produced, and subcontracting. The balance equations among the whole production and inventories, total demand and backlogs are established through constraints (5). Clearly, constraints (5) may easily be modified to accommodate alternative assumptions concerning lost demand. Constraints (7) are constraint conditions, stating that the amount stored must be less than the storage capacity, for each time period. Similarly, constraints (8) stipulate an analogous condition for subcontracted units limited by management. Upper limits on overtime labour capacity are given by inequalities (9).

Table 14-2. S&OP parameters

Period t	1	2	3	4	5	6	7	8	9	10	11	12
N_t	19	9	21	22	21	16	22	20	22	20	20	20
N_t^*	24	17	30	31	30	24	30	28	31	30	20	30
D_t	1475	510	1655	1320	1757	1210	1603	1475	1320	1685	1199	1782
S_t^*	300	300	300	300	300	300	300	300	300	300	300	300
I_t^*	195	195	195	195	195	195	195	195	195	195	195	195

5.2 S&OP optimisation

Working out the problem and using of EXCEL® solver determine the optimal solution according to the given conditions. The results are shown below (Figure 14-3 and Figure 14-4).



Sales & Operations Plan

Parameters					
Beginning Inventory	100	T	Carrying cost	190	€/T
Capacity	65	T/d	Overtime over-cost	700	€/T
			Lost production over-cost	1300	€/T
			Backlog cost	2300	€/T
			Subcontracting cost	600	€/T

Results							
Period *t*	**1**	**2**	**3**	**...**	**11**	**12**	
Demand D_t	1 475	510	1 655	...	1 199	1 782	**16 991 T**
Production	1 235	585	1 365	...	1 300	1 381	**15 161 T**
Subcontracting S_t	140	0	215	...	0	300	**1 730 T**
Total prod P_t	1 375	585	1 580	...	1 300	1 681	**16 891 T**
Inventory I_t	0	75	0	...	101	0	**T**
Production in overtime O_t	0	0	0	...	0	81	**81 T**
Overcost	84	14	129	...	19	237	**1 170 k€**

Figure 14-3. Optimised S&OP results

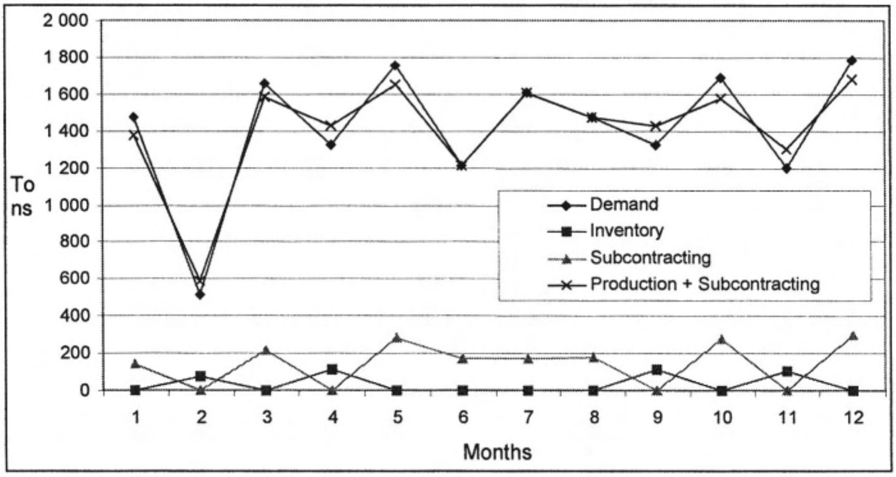


Figure 14-4. Optimised S&OP graph

The recommended strategy is inventory building during the under-load periods and the use of subcontracting in overloaded periods. The demand being strong during period 12, 81 tons are produced in overtime, less expensive than subcontracting and storage during period 11. The over-cost of this scenario is 1 170 K€. If the global approach is logical, the specific choices for each month cannot be obvious [1].

Let us suppose now that various events occur in production few hours, few days after the implementation of this S&OP scenario.

5.2.1 An exceptional order

This order consumes 50% of the beginning inventory! How does the optimum evolve?

Volume in subcontracting will be more significant involving an additional over-cost of 30 K€. The other part of the scenario remains identical (Figure 14-5).

This exceptional order is profitable only if its margin is higher than these 30 K€. In addition, if this event is anticipated, that makes it possible to warn the subcontractor to take the 50 additional tons. If this order is not anticipated, the 50 tons are made in overtime instead of sub-contracted. 5 additional K€ (50 X (700-600)) have to be added.

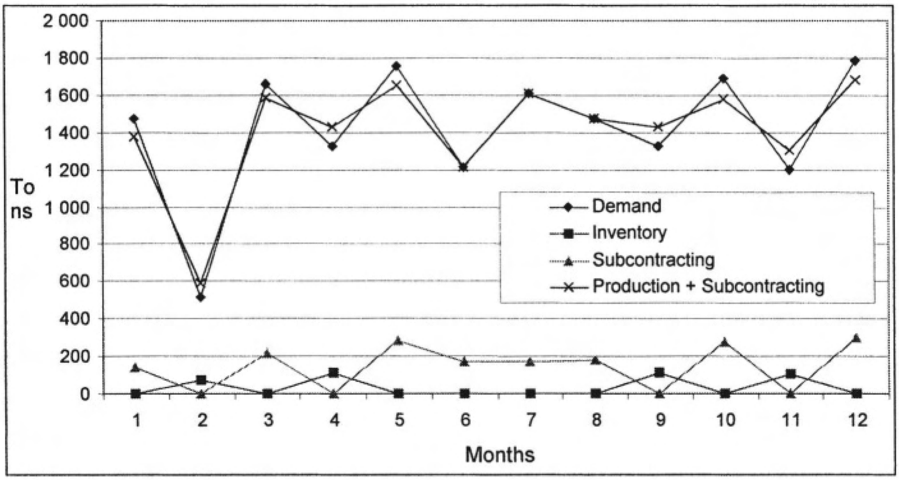


Figure 14-5. Optimised S&OP graph after order integration

5.2.2 Capacity restriction of our supplier

Because of a contract for a new strategic market, our subcontractor can treat only 80 % of our needs! What becomes of the optimum?

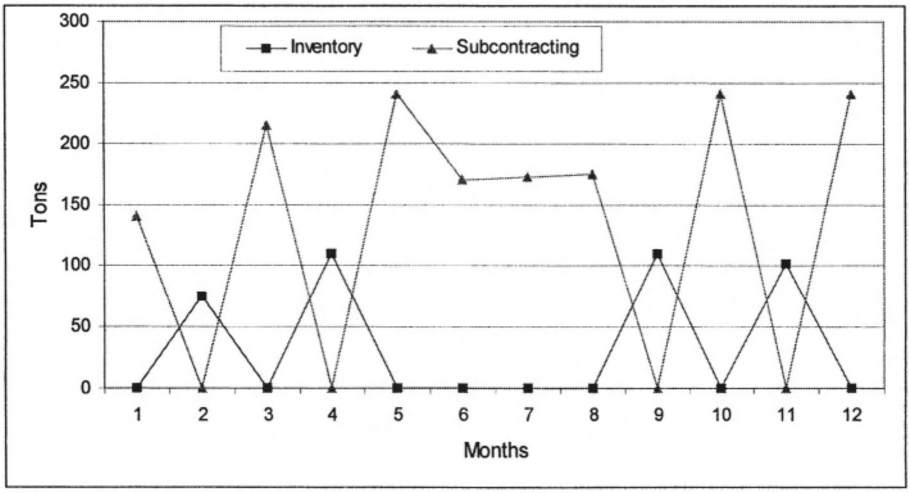


Figure 14-6. Optimised S&OP graph after restriction

Subcontracting is now limited to 240 tons per month (Figure 14-6). Additional days are to be envisaged for the overload periods. The resulting over-cost is of 14 K€.

5.2.3 Reduced capacity

An event in production constraints VPE to produce during the first two months with a reduced capacity of 20 %! Where is the new optimum?

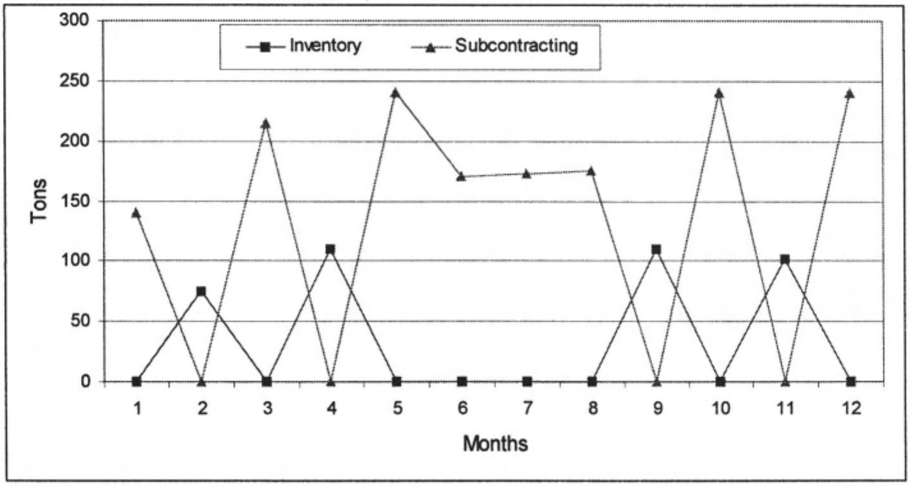


Figure 14-7. Optimised S&OP graph after event

Overtime are carried out in period 1. Subcontracting makes it possible to fulfil demand during months 2 and 3. Then the scenario remains identical (Figure 14-7).

5.2.4 Forbidden Storage

Storage becomes impossible – $I_t^* = 0$. What becomes the optimum?

The under-load periods cause lost in production time and the overload periods imply the subcontracting and overtime saturation. The over-cost of the strategy is then 1 875 K€, that is to say a variation compared to the optimum of 60 % (Figure 14-8)!

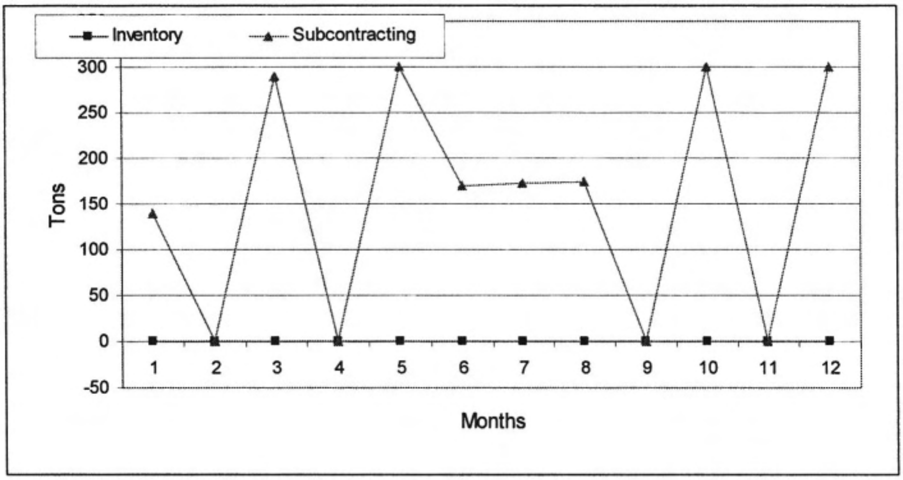


Figure 14-8. Optimised S&OP graph with 0 storage capacity

6. LIMITS OF LINEAR PROGRAMMING

How can we conclude of the preceding simulations? That the level of beginning inventory is not very significant? Certainly, but it is more interesting to note than the optimum is not stable!

Linear Programming makes it possible for the Supply Chain Manager to generate the optimal scenario for a given set of parameters. It is more adapted than the graphic techniques to exploit problems with multiple constraints by providing a technique that leads to an optimal mathematical solution, for a given set of conditions.

However, the scenarios developed previously show that an event in production can make highly diverge the optimum and thus to change the optimal scenario. In a dynamic mode, nothing prevents the model to give highly different solutions at only two days intervals. On a given date, the firm directs its strategy towards the optimum, implements heavy actions (an investment for example), the following day, the conditions are different and the actions differ widely! The dispersion of the mathematical optimum given by the linear programming model is caused by the parameters variability in time whereas they are regarded as static in the model. This assumption is not always exactly satisfied in practice on the whole time-fence of the S&OP: for example, subcontracting capacity can be punctually limited.

Use of such tools cannot be done without paying special attention of the significance and validity of these assumptions, which in practice are unfortunately questioned. Indeed, the coefficients of resources consumption or the costs generally depend on the quantities: a subcontractor reduces his unit price if the quantities are more significant. Linear programming cannot treat these cases. "Linear Programming allows convex structures of production and storage costs (non-decreasing marginal costs). What is awkward in this type of constraint is the impossibility of introducing a launching cost, because in general, at on a few months term, the production is realised at non-decreasing marginal cost, once production of the first unit released" [3]. A problem formulation in linear programming language requires lots of assumptions (linearity and independences of the variables).

7. FUTURE PROSPECTS OF S&OP

How can the models that use Linear Programming be extended? The answer will be to seek models using the non-linear and dynamic programming. Certain authors suggested, in particular cases, other approaches (stochastic optimisation models, Monte Carlo) [7, 10]. How can the S&OP be approached in the industrial case of multiple production lines? A deep research in the scheduling techniques must provide a suitable answer. How can the scenario suggested by the linear program be reinforced? The answer is by using the S&OP ...

The S&OP is the process that drives the capacity level and actions to be implemented so that Master Production Schedules can be completed. It must thus be relatively stable, because of the weight of the taken decisions. For example, the inventory level is limited by the storage capacity. In case of events, the scenario of replacement can become then extremely expensive for the firm whereas a simple contract with a storage partner could be signed to provide a more robust scenario.

The S&OP becomes the decision tool to make the suggested scenario robust by using the model according to the given set of conditions (parameters). The Decision Support System must give an average answer that will be the best possible response for several sets of conditions. The S&OP will not be then mathematically optimised but will vary little when a change in parameters will occur.

Design plans are the traditional tool used to establish the robustness of the system answer (i.e. reduce the variability of the answers) by influencing the control parameters (of adjustment) [9].

In the case of our S&OP models, the input parameters represent the Sales forecasts. The control parameters are those that make possible to control the system: costs of overtime, subcontracting, maximum inventory ... They are the action levers for the planner. The parameters of disturbance are all those which intervene on the system independently of the will of the planner. It is the case for example for the beginning inventory, the capacity of the line ...

Taguchi calls the not-controllable parameters “noises” [11]. The more “robust” a system will be, the lower the variability in scenario. A scenario will be robust if it is not called in question by non-controlled external factors (noises).

Use of Design plans allows to test, with a restricted number of trials, the average scenarios that optimise the S&OP while limiting their variability. The planner must fix, during the process, the nominal values of the control parameters according to a double optimisation:

- the optimum operation of the system,
- the resulting robustness.

We neglect too often the second optimisation. We then work out strategies on paper that cannot to be implemented or lead to the sub-optima in a “disturbed” environment.

8. CONCLUSION

Within the framework of recent work, we have initiated this way of research on the robustness of the S&OP thanks to simulations carried out by taking into account controllable and not-controllable variables for the industrial system and the manager [12]. Our survey of the literature made clear to us that others mathematical tools can be interesting to test these scenarios. We want to show that an optimum research in this field must be obtained by various levels of simulations: a first step to define an optimal target, a second to define a tolerable range of variation without degradation of cost and finally a law defining the marginal loss according to the difference to the optimum.

REFERENCES

1. Crandall R.E., 1998. Production Planning in a variable Demand Environment, *Production and Inventory Management Journal,* 4th Quarter, APICS, 41-48.
2. Genin P., 2000. Le Plan Industriel et Commercial : Recherche des pratiques optimales du processus d’élaboration, *Mémoire de DEA,* LATTS, Ecole Nationale des Ponts & Chaussées, Paris, France.
3. Giard V., 1988. *Gestion de la Production,* Economica, Coll. Gestion, 2ème édition.
4. Giard V., 1998. *Processus productifs et Programmation linéaire,* Economica.
5. Heizer and Render, 1995. *Production and Operations management,* Prentice Hall.
6. Lamouri S and Thomas A., 1999. Optimization of the process of development of the SOP, In: *Proceedings of the International Conference on Integrated Design and Production* (CPI' 99), Tanger, Maroc, 328-338.

7. Moutaz K., 1998. An aggregate production planning framework for the evaluation of volume flexibility, *Production Planning and Control, 9* (2), 127-137.

8. Nollet, Kelada and Dioro, 1992. *La gestion des opérations et de la production,* Editions Morin Gaetan.

9. Pillet M., 1992. *Introduction aux plans d'expériences par la méthode Taguchi,* Editions d'organisation Université Paris.

10. Silva Filho O. S., 1999. An aggregate production planning model with demand under uncertainty, *Production Planning and Control,* 10 (8), 745-756.

11. Taguchi G., 1987. *Orthogonal arrays and linear graph,* American Supplier Institute press.

12. Thomas A. and Lamouri S., 2000. The new problem with Sales, Inventories and Operations planning in a Supply Chain environment, In: *Proceedings ofthe International Society for Optical Engineering on Intelligent Systems in Design and Manufacturing,* Boston, USA, 321-329.

13. Vollmann, Berry and Whybark, 1992. *Manufacturing, Planning and Systems Control,* The Business One Irwin.

Chapter 15

RESPONSE SURFACE-BASED SIMULATION METAMODELLING METHODS

with applications to optimisation problems

Galina Merkuryeva

Abstract: Response surface-based simulation metamodelling procedure is presented. It is aided by a computer generated experimental designs, automatic control of simulation experiments and sequential optimisation of the metamodels fitted to a simulation response surface function. The metamodelling procedure is applied to optimisation of a shop-floor production level.

Key words: simulation, metamodelling, response surface, optimisation.

1. INTRODUCTION

The complexity of the simulation studies increases with the complexity of the real system. There is a need for methods and metamodels that help to analyse the behaviour of complex models itself, to improve their validation process, to make easier an analysis of the simulation output data and to aid in optimisation of the simulation model. Response surface methodology (RSM) is a collection of statistical and mathematical techniques for optimisation of stochastic functions [2]. RSM is developed in order to analyse experimental data and to build empirical models based on observations of the stochastic function. In case of a computer simulation model simulation output dependence on its input variables could be interpreted by a response surface function. Main advantage of RSM methodology is its applicability for a small number of observations in conditions of costly and time-consuming simulation experiments.

2. RESPONSE SURFACE METHODOLOGY

In the paper response surface methodology is used to approximate a stochastic function of 'input-output' relations of a simulation model in order to optimise it by supporting simulation experimental design and simulation response optimisation. For simplification, all independent and dependent variables are supposed to be numerical and measured on a continuous scale. Let the simulation dependent variable or response is y and $(x_1,\ldots,x_i,\ldots,x_n)$ are independent variables or factors. Then the response surface function f could be presented in the form:

$$y = f(x_1,\ldots,x_j,\ldots,x_q) + \varepsilon, \tag{1}$$

where q is a number of factors, and the term ε represents a statistical error, often assuming it to be independent and to have a normal distribution with mean zero and constant variance.

Graphical representation of response surfaces plays an important role in RSM methodology and is given in three dimensions for a response y above the x_1 and x_2 space, and as a contour plot $y = f(x_1, x_2)$ on the plane x_1x_2 with contour lines of constant responses. The following are typical response surfaces contour plots: 1) Mound-shaped response surface, that has elliptical contours with a stationary point in a point of a maximum response; 2) Saddle–shaped response surface, that has a hyperbolic system of contours with a stationary point that is neither a maximum nor a minimum point and called as a saddle point; 3) Constant (stationary) ridge response surface, which contours present concentric greatly elongated ellipses with a stationary point in the region of experimental design; the stationary point is a point of a maximum response, but yet there is essentially a 'line maximum' within the design region design; and 4) Rising (or falling) ridge response surface with the stationary point that is a maximum response and could be remote from the design region.

Usually, the form of the true response function is apriori unknown and should be approximated. So, successful use of RSM methodology is critical to development a suitable approximation for function f that could be very complicated. Usually, low-order models, either a first-order or second-order models, are used to explore the behaviour of response surface in some relatively small region of the independent variables. Experimental designs, such as the first-order orthogonal designs, central composite designs (CCD) or spherical designs for fitting response surfaces, are developed within RSM methodology. Response improvement with a steepest ascent provides sequential optimisation of a local linear model. In the case of quadratic approximations canonical or ridge analysis should be applied.

Simulation metamodelling that is based on the RSM methodology supposes: 1) an experimental analysis of independent variables or factors and their influence on the behaviour of the real system or process to be analysed; 2) statistical modellling to fitting simulation response surface approximations in small subdomains of independent variables or factors; 3) exploration and optimisation of response surface approximation functions. The simulation model is supposed to be in a steady state.

3. SIMULATION METAMODELLING PROCEDURE

Optimisation of the response surface function comprises two phases (Figure 15-1). In the first phase the response surface function is fitted by a first-order model in the small subdomain far from the response stationary point, and a line search in a direction of a simulation response improvement is performed step-by-step. In the second phase the response surface function is approximated by a second-order model within the stationary region and its canonical analysis is performed.

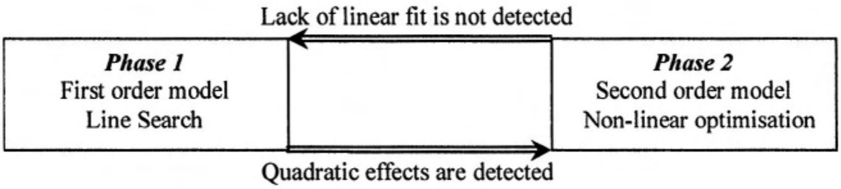


Figure 15-1. Two phases of the response surface function optimisation

Simulation metamodelling based on the response surface methodology presents a sequential procedure. In each iteration m a small subdomain or region of experimentation is described by the q-dimensional rectangle $[t_1^m, u_1^m] \times \ldots \times [t_q^m, u_q^m]$ with a central point ξ_j^m and a step c_j^m, defined by

$$\xi_j^m = \frac{t_j^m + u_j^m}{2}, \quad c_j^m = (u_j^m - t_j^m)/2, \quad j = 1,\ldots,q. \qquad (2)$$

Simulation metamodelling procedure based on the response surface method includes the sequence of following main steps [3].

1. Local approximation of the response surface function by a first-order model $y = a_0 + \sum_{j=1}^{q} a_j \xi_j + \varepsilon$ in the current region of experimentation or interest. To increase the numerical accuracy in estimation of regression coefficients factors or natural input variables are coded by a formula:

$$x_j = \frac{\xi_j - \xi_j^m}{c_j^m}, \; \xi_j = c_j^m x_j + \xi_j^m, \; j = 1,...,q. \qquad (3)$$

The coded first order model $Y = \beta_0 + \sum_{j=1}^{q} \beta_j x_j + \varepsilon$, with coded variables x_j, is evaluated by using two-level full factorial experimental designs added by simulation experiments replicated in the central point.

2. Checking the fit of a first-order metamodel to describe behaviour of the simulation response in the current region of interest. Lack-of-Fit test using the p-values based on ANOVA table as well as residual analysis and influence diagnostics should be performed in order to check adequacy of a metamodel and to verify the least squares method assumptions. A significant lack-of fit may be a result of factors interactions excluded from the model or unusually large residuals not well explained by the fitted approximation. If it is occurs on the initial phase of the metamodelling procedure and probably far from a stationary point, then it could be reasonable to decrease the size of the current experimental region in order to fit a first-order approximation model. Otherwise, fitting a second-order model may decrease the efficiency of an optimisation algorithm.

Moreover, in the case of model adequacy, direction of significant improvement for a simulation model response could not be easily found, if estimates of regression coefficients received are quite small comparing with an estimation error or even they are close to 0. In these cases increasing a length of a simulation run or a number of replicate runs for each experimental point may guarantee statistical significance of the ascent direction on the step 3.

3. A line search in the steepest ascent direction (or steepest descent direction in the case of function minimisation). A line search is performed from the central point of the current experimental region in the steepest ascent direction defined by a vector $(b_1,...,b_j,...,b_q)$, where b_j is estimation of β_j. The increments $(\Delta_1,...,\Delta_q)$ along the projection of the search direction could be calculated by a formula for coded factors taking account their main effects:

$$\Delta_j = \frac{-b_j}{\max_i |b_i|}, j = 1,...,q. \quad (4)$$

The *p*th line search point on *m*th iteration or ascent trajectory with an initial point $(\xi_1^m,...,\xi_q^m)$ is given by $(\xi_1^m + p\Delta_1 c_1^m,...,\xi_q^m + p\Delta_q c_q^m)$. Different line search stopping rules, such as straightforward rule, n-in-a-row stopping rule, recursive rule, are based on identification of conditions when no further response improvement is observed.

4. Local approximation of response surface function by a second-order model and its testing for adequacy. General coded model includes quadratic effects of factors and their pair interactions defined by additional regression parameters β_{jj}, β_{ik}, correspondingly:

$$Y = \beta_0 + \sum_{j=1}^{q} \beta_j x_j + \sum_{j=1}^{q} \beta_{jj} x_j^2 + \sum_{i,k:i \neq k} \beta_{ik} x_i x_k + \varepsilon. \quad (5)$$

CCD design is recommended for fitting the second-order model, as it could be easily restructured from the factorial designs used for estimating a first-order model. It is also could be transformed in the orthogonal plan by adding replicate experiments in the central point of the current experimental region.

5. *Canonical analysis to define the location and nature of a stationary point of the second-order model.* If the second-order model is found to be adequate, then canonical analysis based on signs of eigenvalues $\lambda_1, \lambda_2,..., \lambda_q$ (see [2]) allows to determine the behaviour of the simulation response close to the stationary region. If all eigenvalues are positive or negative, the stationary point is a point of minimum or maximum response, or if eigenvalues are mixed in signs, then the stationary point is a saddle point which is neither a point of estimated minimum or maximum response. If a stationary point that is a point of maximum or minimum response is received outside the experimental region where the model is not reliable, then ridge analysis of the response surface should be performed.

In the case of stochastic function optimisation when several responses or factors are involved, the desirability function method could be very useful. The method makes use the desirability function in which researcher's own priorities on the response values could be included into the optimisation procedure. The method in a graphical and interactive mode is realised by Minitab 13.2 *Response Optimizer* that allows to get the response optimal value and to make its sensitivity analysis to setting the initial optimisation conditions.

Let *L* and *U* define lower bound and upper bound values for the response, *T* corresponds to the response target that could be interactively fixed by a researcher. In the case of the response maximisation the desirability function *D(Y)* is defined as:

$$D(Y)=\begin{cases} 0, & Y<L, \\ ((Y-L)/(T-L))^{w}, & L<Y<T, \\ 1, & Y>T. \end{cases} \qquad (6)$$

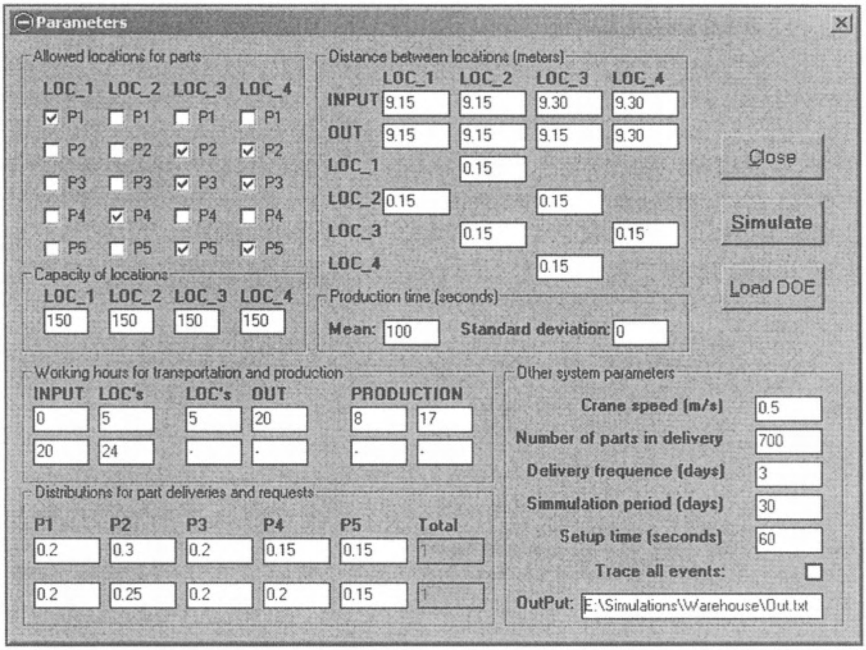
In (6) weight *w* defines the type of the desirability function. In the case of the response maximisation and the weight values $w = 1$, w=[0.1,1)or $w = (1,10)$, function $D(Y)$ could be linear, concave or convex, so that the desirability quantity increases linearly, or this quantity could be quite high even far or only close to the target.

In the most RSM applications available in literature, it is used to end an optimisation procedure after fitting only once second-order function approximation. Other stopping strategies that look more reliable are discussed in [3].

4. OPTIMISATION OF A SHOP-FLOOR PRODUCTION LEVEL

The application problem could be described as follows [1]. The shop-floor is processing wooden plates of different types (dimensions, surface, colour). Plates are delivered into a store area once or twice per week and are located in the storage area. The number of plates of different types in one delivery may vary and is defined by an empirical probability distribution. The number of store locations is less than the number of the plate types. Each location could contain a stack of either the same type or different types. The plates are taken out every day according to the production requirements. Plates are moved in and out of the storage area one by one by the same crane. The production demand may vary from the day to day but monthly demand however stays fairly stable. Crane operations and production process are scheduled in advance. A distance matrix defines locations sizes, as well as distances between locations, between locations and store area's input/output buffers. Store location assignment to different plate types is fixed in advance.

The simulation model is designed using Visual Basic 6.0 development environment and aimed to simulate transportation and production operations in the shop-floor storage and production areas. The simulation model consists of three logical blocks that implement input, simulation itself and simulation output procedures. Input procedures provide setting of variables and probability distributions, as well as loading the experimental design file (Figure 15-2). Simulation procedures control crane activities, input and output buffers and location conditions, production time and parts delivery.



Parameters

Allowed locations for parts

LOC_1	LOC_2	LOC_3	LOC_4
☑ P1	□ P1	□ P1	□ P1
□ P2	□ P2	☑ P2	☑ P2
□ P3	□ P3	☑ P3	☑ P3
□ P4	☑ P4	□ P4	□ P4
□ P5	□ P5	☑ P5	☑ P5

Capacity of locations

LOC_1	LOC_2	LOC_3	LOC_4
150	150	150	150

Distance between locations (meters)

	LOC_1	LOC_2	LOC_3	LOC_4
INPUT	9.15	9.15	9.30	9.30
OUT	9.15	9.15	9.15	9.30
LOC_1		0.15		
LOC_2	0.15		0.15	
LOC_3		0.15		0.15
LOC_4			0.15	

Production time (seconds)

Mean: 100 Standard deviation: 0

Working hours for transportation and production

INPUT	LOC's	LOC's	OUT	PRODUCTION	
0	5	5	20	8	17
20	24	-	-	-	-

Distributions for part deliveries and requests

P1	P2	P3	P4	P5	Total
0.2	0.3	0.2	0.15	0.15	1
0.2	0.25	0.2	0.2	0.15	1

Other system parameters

Crane speed (m/s) 0.5

Number of parts in delivery 700

Delivery frequence (days) 3

Simmulation period (days) 30

Setup time (seconds) 60

Trace all events: □

OutPut: E:\Simulations\Warehouse\Out.txt

Figure 15-2. Simulation model interface

Output procedures create different output files for a further analysis. Special input procedures provide possibility for automatic run of simulation experiments using predefined design of experiments from the file (Figure 15-3). Simulation model interface provides input variables set up form and simulation process interface.

The simulation metamodelling is aimed to enhance a simulation model optimisation problem based on RSM methodology. Response surface-based metamodelling procedure was realised within Minitab 13.2 Statistical software aided by automatic control of simulation runs. A response surface has been explored to predict the maximum production level. After screening experiments it has been concluded that the production level is mainly affected by the crane speed (X_1, m/s) and production average time per unit (X_2, s).The domain of independent variables has been defined as follows: $0.5 \leq X_1 \leq 3.5$, $10 \leq X_2 \leq 230$. Long simulation runs equal to 30 days were performed in the study.

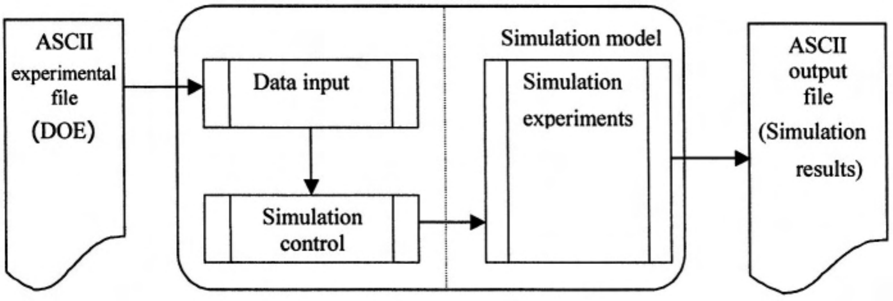

ASCII experimental file (DOE)
Data input
Simulation control
Simulation model
Simulation experiments
ASCII output file (Simulation results)

Figure 15-3. Automatic run of simulation experiments

Step 1. The initial experimental region was defined by intervals $X_1 = [0.5, 1.5]$ and $X_2 = [50, 150]$ with the central point: $x_{10}^{1} = 1.0\text{m/s}$, $x_{20}^{1} = 100\text{s}$. 2^2 full factorial design was used added by three replicate experiments in the central point to estimate fit-of-lack and to increase reliability of estimating main effects of the factors. The simulation experimental results are presented in the Table 15-1.

Table 15-1. Results of simulation experiments

Coded values		Natural values		Simulation response
x_1	x_2	X_1 – speed	X_2 – time	Y (prod)
-1	-1	0.5	50	201
-1	1	0.5	150	200
1	-1	1.5	50	227
1	1	1.5	150	212
0	0	1.0	100	227
0	0	1.0	100	215
0	0	1.0	100	204

Figure 15-4 shows analysis of variance of the first-order model. Testing lack-of-fit gives the *p*-value $p = 0.730$ implying the resulted linear model to be adequate. Its contour plot and the steepest ascent direction are shown in Figure 15-5. According to (4), increments in this direction for coded and uncoded factors are equal to: $\Delta x_1 = 1$, $\Delta x_2 = -0.42$, and $\Delta_1 = 0.5$ m/s, $\Delta_2 = -21$ s (in negative direction). The step size is defined as $\Delta = (0.5, -21)$.

```
Source          DF   Seq SS   Adj SS   Adj MS     F      P
Regression       2   425.00   425.00   212.50   2.35  0.212
Linear           2   425.00   425.00   212.50   2.35  0.212
Residual Error   4   362.43   362.43    90.61
Lack-of-Fit      2    97.76    97.76    48.88   0.37  0.730
Pure Error       2   264.67   264.67   132.33
Total            6   787.43
```

Figure 15-4. Analysis of variance of the first-order model

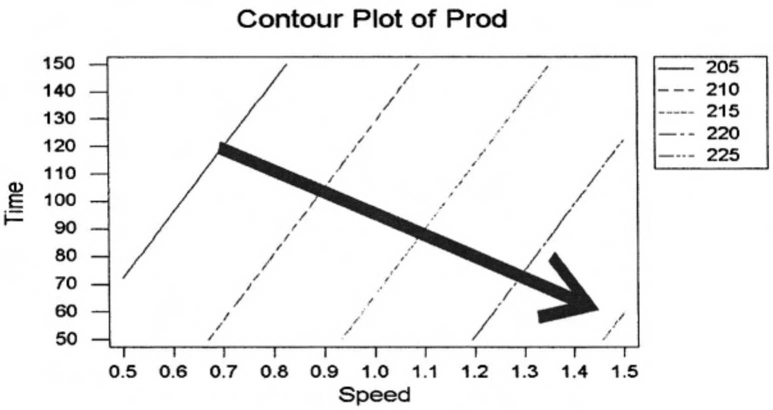


Figure 15-5. First-order model contour plot at the first step

Results of simulation runs for the region central point and two consequent line search iterations are given in Table 15-2. The response-improved value has been found on the first iteration in the point (1.5, 0.79). According to straightforward stopping rule, this point has been fixed as the center point of a new experimental region. The number of simulation experiments required at the fist step was equal to 9.

Table 15-2. Simulation response values in the line search process

	X_1	X_2	x_1	x_2	Y
X_0	1.0	100	0	0	204
$X_0+\Delta x$	1.5	79	1	-0.42	228
$X_0+2\,\Delta x$	2.0	58	2	-0.84	226

Step 2. The behavior of the simulation response in a new experimental region is defined by [1,2]×[29,129] with the central point (1.5, 0.79) has been approximated by a first-order model $\hat{y} = 221 + 8.2x_1 + 2.7x_2$. A line search step size in the steepest ascent direction was determined as $\Delta_1 = 0.5$ m/s, $\Delta_2 = 16.5$ s. Nevertheless, results of a linear search didn't provide essential response improvement. The number of simulation experiments required at the second step was equal to 11.

Step 3. Local approximation of the simulation response surface by a second-order model in the experimental region of independent variables, i.e. [1,2]×[29,129], defined at the second step, has been performed. Computer generated CCD design included 13 experiments with 5 replicate runs in the central point of the experimental region. The local quadratic model was defined by the following regression function:

$$y = 224 + 11speed - 7.8time - 6.3speed^2 - 3.8time^2 - 8.2speed{*}time\,.$$

ANOVA analysis allowed to conclude that lack-of-fit of quadratic model was not statistically significant. In particularly, p-value 0.626>0.05. Residual analysis proved that none of the least square assumptions were violated. Influence diagnostics did not identify outliers and potentially highly influential observations as standardised residuals were quite small, h_{ii} values did not exceed their critical value 2 k/n = 0.92 and the corresponding value of Cook statistics was less then 1. In particularly, LOF p-value 0.626>0.05. Two additional experiments were performed.

Step 4. Based on the quadratic model contour plot (Figure 15-6) the confidence region for exploring the optimal response was defined by the lower and upper bound values, 225 and 228 units, correspondingly. The experimental region of factors has been defined by CCD design. After quadratic response model optimisation using linear desirability function, the optimal predicted response value of 228 units has been received in the satisfactory region, i.e. in the point (1.5, 79.0).

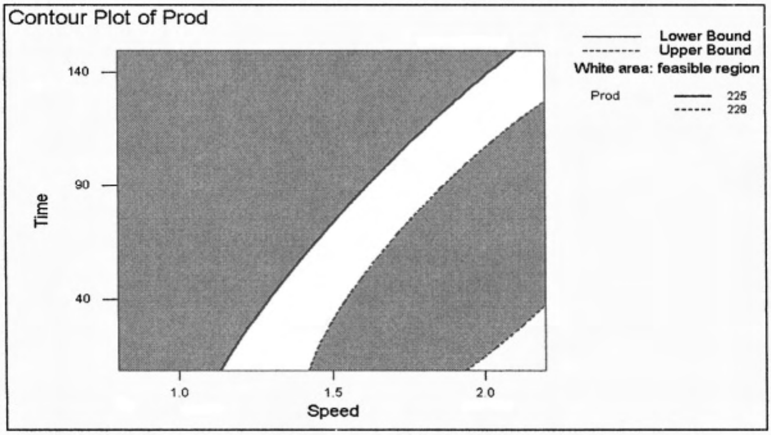
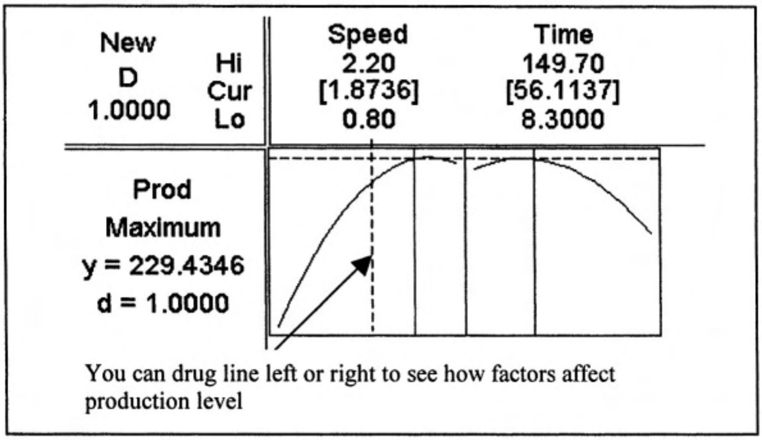


Figure 15-6. Quadratic model detailed contour plot

Sensitivity analysis performed by independent variations of initial optimisation conditions allowed improving the response optimal value till 229 units (Figure 15-7). The crane transportation speed and production time per unit equal to 1.87 m/sec and 56 sec, correspondingly. In total 33 simulation experiments were required to perform metamodelling procedure in order to predict a maximum production level.



New D 1.0000		Speed	Time
	Hi	2.20	149.70
	Cur	[1.8736]	[56.1137]
	Lo	0.80	8.3000

Prod
Maximum
y = 229.4346
d = 1.0000

You can drug line left or right to see how factors affect production level

Figure 15-7. Sensitivity analysis of the optimal response value

5. CONCLUSIONS

Application of metamodelling provides the better understanding of the resulted complex stochastic simulation model substituting or enhancing the later with simpler and less costly analytical model. Placing the metamodel within the simulation output analysis could provide enhanced exploration of the modelling results with the goals of the model optimisation and interpretation of its output.

REFERENCES

1. Merkuryeva G., 2001. Performing simulation studies including metamodelling. In: *Proceedings of the International Conference on Transformation of Economic and Social Relations: Processes, Tendencies and results,* Turiba, 2001, 265 - 270.
2. Myers R. H. and Montgomery D. C., 1995. *Response Surface Methodology: Process and Product Optimisation Using Designed Experiments* / Raymond H. Myers and Douglas C. Montgomery, John Wiley & Sons, N.Y.
3. Neddermeijer H.G., G.J. van Oortmarssen, Piersma N. and Dekker R., 2000. A Framework for response surface methodology for simulation optimization. In: *Proceedings of the 2000 Winter Simulation* Conference / Eds. J.A. Joines, R.R. Barton, K.Kang and P.A.Fishwick, 129-136.

This page intentionally left blank

PART III: DECISION AID TOOLS

This page intentionally left blank

Chapter 16

A MODELING AND SIMULATION FRAMEWORK FOR SUPPLY CHAIN DESIGN

Hongwei Ding, Lyès Benyoucef, Xiaolan Xie

Abstract: A modeling and simulation framework to facilitate supply chain design is presented. Based on the modeling framework, a discrete event simulation package is developed to achieve modeling flexibility and simulation efficiency. Typical attributes and behaviors of each supply chain facility are defined and integrated into building blocks of the simulation package, with which practitioners can easily build a supply chain model, simulate it with different operation rules and precisely evaluate its performances. A case study of automotive industry is presented and a distribution network model is proposed.

Key words: supply chain modeling, distribution network, discrete-event simulation, make-to-stock, make-to-order.

1. INTRODUCTION

In the global competitive market, Supply Chain Management (SCM) receives more and more attention and interest in both industrial practice and academic community. Traditionally, different suppliers, manufacturers and distributors along the supply chain operate independently. Each individual facility has its own operation strategy and aims to optimize its own profit with almost no regard to how its decision impact other partners. Lack of coordination and information sharing results in high operation costs and inefficiency. SCM is thus proposed as a set of approaches to efficiently integrate suppliers, manufacturers, warehouses, and stores, so that merchandise is produced and distributed at the right quantities, to the right locations and at the right time, in order to minimize system-wide costs while satisfying service level requirements [9].

Generally speaking, the model of a supply chain network depends on practitioner's objective. For example, when we focus on the design of a production distribution network, more attention is paid to operations within manufacturers and distributors. Of course, the relationship between suppliers and buyers should be carefully modeled and studied when dealing with supplier selection problem. In order to achieve modeling flexibility, in this chapter, we propose a generic modeling framework that clearly defined basic supply chain facilities such as supplier, manufacturer, distributor and etc. Practitioners will benefit from the consistent modeling framework when dealing with various supply chain network design problems.

Processes related to SCM range from the very simple to the very complex. Complex processes are very difficult to model, analyze and optimize using standard mathematical methods. Simulation is an alternative method of analysis that offers numerous benefits. Provided a supply chain network, simulation performs relatively precise evaluation of lead time, inventory costs and so forth, with respect to various stochastic aspects, such as demand fluctuation and transportation risks. Following the modeling framework, we also developed a discrete-event simulation package to facilitate performance evaluation. This allows us to build supply chain network model and retrieve its performances through simulation quickly.

The rest of the chapter is organized as follows. Section 2 reviews literatures on supply chain modeling and discrete event simulation. Our generic modeling and simulation framework for supply chain network design is presented in Section 3. Section 4 is dedicated to an automotive distribution case study, where the network model and corresponding simulation implementation are described in details. The case study is introduced to illustrate the modeling and simulation processes by use of the proposed framework. Numerical results are provided and comparisons of alternative distribution scenarios are given. Finally, conclusions and future research work are addressed.

2. LITERATURE REVIEW

There are numerous papers dedicated to supply chain modeling and simulation-based evaluation. In this section, we briefly summarize some important works dealing with these problems.

Cohen and Lee [3] proposed a literature review of past attempts, introduced a modeling framework and an analytic procedure for evaluating the performance of supply chain networks. They stated that the distribution network plays a critical role in improvement performances of the whole supply chain, since it is the segment to face the final customers directly. Cohen and Moon [4] extended their research result to an optimization model of supply chain. Due to the deterministic characteristic of the model, they did not investigate the impact of dynamic behaviors, such as demand fluctuation and other kinds of uncertainty. Geoffrion and Powers [5] present a comprehensive literature review of distribution network design. Different approaches for distribution network design, such as linear programming and simulation, are reviewed and compared in an evolutionary perspective.

A number of modeling frameworks are proposed in the past. Slats *et al.* [10] stated that logistic chain modeling is a decisive factor in increasing the performances of a global supply chain. Breitman and Lucas [2] proposed "PLANETS" as a framework for a model of a production-distribution system to determine what products to produce, where and how to produce them, which markets to pursue and what resource to use. Stock *et al.* [11] described a conceptual framework to take into consideration logistics, strategy and organizational structure.

In general, analytical models are highly simplified models of business processes in order to make the models solvable. To obtain very accurate and detailed models, one has to take into account many realistic features in the context of supply chain management. Due to different kinds of uncertainty and risk and complex business processes, a simulation approach is preferred to analytical methods. Bhaskaran [1] presented a simulation analysis of the supply chain instability. The impacts of various factors that amplify the dynamics and instability have been studied. Petrovic *et al.* [7] and Petrovic *et al.* [8] proposed several fuzzy models of supply chains and a corresponding simulator to assist decision-making in operational control of supply chains. Two sources of uncertainties, customer demand and external raw material supply are considered and represented by fuzzy sets. The main assumptions include serial supply chains and single product type. Jansen *et al.* [6] proposed a simulation model to evaluate logistic and financial performances of various alternative logistic scenarios for a multi-compartment distribution network in the catering supply chain.

Many of the previous models are either only a conceptual network, and therefore too abstract to realize, or only designed and simulated for a specific problem, and hence lack general applicability. There is a need for a general approach for both supply chain modeling and its evaluation by simulation.

3. A SUPPLY CHAIN MODELING AND SIMULATION FRAMEWORK

A supply chain is generally viewed as "a system whose constituent parts include material suppliers, production facilities, distribution services and customers linked together via the feed forward flow of materials and the feedback flow of information" [12]. In this framework, a supply chain comprises a number of interconnected entities and transportation links. Note that many existing works only considered supply chains of serial configuration. To be realistic, we represent the structure of a supply chain as a network (Figure 16-1), which consists of nodes representing supply chain entities and arcs representing transportation links and information links.

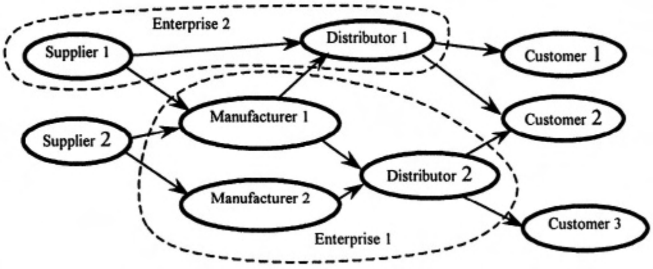


Figure 16-1. Supply chain network structure

Since we focus on supply chain performance evaluation at the strategic level, necessary assumptions are introduced to simplify and abstract key processes. Fundamental parameters of each facility are attributes of corresponding building blocks and represent aggregated performance measures of corresponding entities. Operation rules are integrated to describe the dynamic behaviors of supply chain entities and to facilitate evaluation of supply chain operation strategies.

Supply chain facilities interact with each other via transportation and information links. In this chapter we study the supply chain activities from an operational point of view. Financial flow is not taken into consideration in this study.

3.1 Supply chain facilities

Typical supply chain facilities are supplier, manufacturer, transit point, distributor and customer. In order to deal with information flow in a more flexible way, an additional facility "Enterprise" is introduced in this framework. All these supply chain entities are described in this section.

3.1.1 Enterprises

An enterprise is a supply chain entity that links and coordinates some geographically dispersed facilities. An enterprise probably consists of the whole supply chain including suppliers, manufacturers, distributors and customers. Or in some other cases, an enterprise owns just several manufacturers and distributors. For example, in Figure 16-1, enterprise 1 comprises two manufacturers and one distributor. Enterprise 2 comprises one supplier and one distributor.

Within this framework, an enterprise takes the role of collecting and analyzing demand and supply information, while transactions related to material flow is excluded from enterprise's functionalities. In particular, an enterprise only handles information flows and makes decisions on enterprise-level issues, such as order assignment, product deployment. Coordination between different enterprises is excluded in this framework and the relations between two enterprises are mainly supplier/customer relations. As a result, supply chain performances are evaluated from the point view of the focal enterprise.

From the point view of an enterprise, there are two categories of demand: internal demand and external demand. Internal demand is the request generated from facilities that belong to the enterprise. External demand is the request issued by other enterprises. In order to achieve the flexibility of information flow modeling, we assume that all demands are received by enterprises, no matter internal or external. For example when a customer generates demand for a final product, corresponding order is forwarded to the enterprise from which the final product is to be delivered. Then the enterprise makes decision to assign the order to self-owned facilities or other enterprises according to its respective operation rules.

Several other assumptions are added to abstract the enterprise concept:

- One facility can only belong to one enterprise. That is to say, there is no intersection between two different enterprises.
- There is no coordination between different enterprises. Decisions are made based on local operation rules.

3.1.2 Suppliers

A supplier is a facility that manufactures, assembles, produces, decorates, imprints or otherwise supplies promotional products for sale to distributors. Supplier provides raw materials, components, semi-finished products and even final products to procurers who make use of them to deliver final products to end costumers. For a supply chain, supplier nodes are usually the source nodes where material flows originate.

In our framework, operation details of supplier are not taken into consideration. The relationship between the focal enterprise and its suppliers is represented in term of supply contract. Typical supply contract concerns several key factors, such as order frequency and minimum order quantity. Correspondingly, relevant attributes of the supplier building block are listed as follows:

- Supply lead time is customer and product dependent. Each supplier contains its own table of supply lead time.

- Product price is not necessarily customer dependent. But the price is typically related to the product type and order quantity. So product price table is also an indispensable attribute.
- Minimum order size is introduced to guarantee the scale of economy.

3.1.3 Manufacturers

A manufacturer is a critical supply chain facility that transforms input raw materials/components into desired output products. Generally speaking, a manufacturer is the place where transformation processes occur.

From the simulation point of view, a manufacturer is considered as a server with finite capacity, each manufacturer comprises a queue of orders that are to be processed. Each manufacturer also holds inventories of raw materials/components for production preparation and finished goods inventories (FGI) in order to form a batch for outgoing transportation.

Important attributes related to this building block include:

- Production capacity
- Production lead time
- Production cost
- Minimum production lot size
- Bill-of-Material

3.1.4 Transit points

A transit point is typically an intermediate node between different transportation modes, at which no stock is purposely built to anticipate future demand. The only goal of a transit point is to distribute products from incoming links to outgoing links as soon as possible. The stock built and the time spent at each transit point basically come from two sources: limited handling capacity of the transit point and difference batch sizes of upstream and downstream links. In the simulation package, a transit point is treated as a capacity-limited server with batch arrivals and batch departures.

Important attributes related to this building block are listed in following:

- Transshipment time
- Operation cost
- Inventory cost

3.1.5 Distributors

Distributors play a critical role in supply chains. One kind of distributor is named as a consolidated distributor if strategic stock is hold inside. According to the current inventory level, customer demands are fulfilled or backordered by the consolidated distributor. Specific inventory policies are employed to balance lead time and lost profits of stock-outs. The handling capacity of a distributor usually depends on the size of the delivery fleet and the delivery frequency. Regular carrier departure is scheduled for each link. Available products in the distributor can be loaded and transported to corresponding customer site. The transportation delays are major sources of randomness of a supply chain network.

Inventory control policies are defined as operation rules of this building block. Several classical inventory control policies, such as base stock, lot for lot, are defined and implemented in the simulation package.

Other attributes related to this building block are listed as follows:

- Storage capacity
- Handling capacity
- Operation cost
- Inventory cost

3.1.6 Customers

Customers are sink nodes of material flows. Demands for different types of products are generated from customers and fulfilled finally by periodic deliveries. Demand quantity of an order follows some random distribution or given sample path with or without seasonality. The demand interval can be fixed or dynamic.

Important attributes related to this building block are listed in following:

- Demand quantity
- Demand frequency
- Expected delivery date

3.1.7 Transportation links

All previous supply chain facilities are involved in a transportation network and connected by transportation links. Detailed model for a transportation link depends on whether the transportation is out sourced. If the transportation is not out sourced, then a transportation link can be considered as a finite capacity queue with the number of servers equal to the number of transportation carriers. For materials to flow throughout the supply chain network, routing decisions are indispensable. In this framework, routing decisions can be made either by simple operation rules or by optimization methods.

The transportation link can be modeled by the following data:

- Transportation delay
- Transportation cost
- Minimum and maximum batch size (truckload, carrier load)
- Delivery frequency

– Link reliability
– Transportation mode and energy consumption

3.2 Discrete-event simulation package

Based on the conceptual network model, a discrete event simulation package written in C++ is developed to facilitate performance evaluation of different network configurations.

A discrete-event simulation environment is developed. It employs event-scheduling method to advance the simulation clock. Moreover, this simulation package contains a number of building blocks to represent basic supply chain entities. Concerning the uncertainties of supply chain operations, a multiple-stream random-number generator is provided, which enables practitioners to evaluate the robustness of specific scenario by introducing different kinds of uncertainties and randomness.

Due to the limitation of the space, details related to simulation implementation are not presented in this chapter.

3.3 Key Performance Indicators

At the stage of strategic configuration, modern management uses Key Performance Indicators (KPIs) to measure the effectiveness of processes in both a quantitative and a qualitative way. In this framework, two categories of indicators are introduced to facilitate performance evaluation.

3.3.1 Logistic indicators

Order-to-Delivery lead time is the time elapsed between the placement of order by customer and the delivery of products to the customer. Normally, lead time includes the time necessary for order processing, the production time and transportation time.

Ratio of on-time delivery is the fraction of customer orders that are fulfilled on time. This ratio is a critical indicator for customer service level, which is basically a function of several different performance indices. *Order fill rate* is the fraction of customer demands that are met from stock. Another measure is the *backorder level,* which is the number of waiting orders.

Inventory position is the sum of on-hand inventory, those units waiting to be produced in manufacturers and units in transit. *On-hand Inventory* represents stock kept in consolidated distributor that is instantly available to fill future demands. *Inventory level* is the quantity of products contained in distributor, composed of on-hand inventory and stock of those products that are already allocated but still hold to form the delivery batch.

Resource utilization is also an important indicator for production planning, resource allocation and so on. For instance, utilization ratio of production capacity for manufacturer is introduced in the simulation model. Different production policies can be compared with respect to this indicator.

3.3.2 Financial indicators

Generally, costs are related to production, transportation, inventory and so forth. There are several kinds of fixed and operational costs associated with a supply chain. Production costs, transportation costs and inventory costs are typical financial indicators.

4. SIMULATION RESULTS

In this section, we present an industry case study. The case study deals with a mixed Make-to-Order (MTO) and Make-to-Stock (MTS) automotive distribution network, where the objective is to evaluate and analyze two different network configurations. Following the framework, each building block of the simulation package is extended to constitute the simulation model.

4.1 A case study of distribution network design

Figure 16-2 is the graphic representation of the reference scenario, namely Decentralized Distribution Network (DDN), including three manufacturers, three transit points (TP1, HB1 and HB2), four distributors and sixteen customers. In particular, facilities covered by shadow area constitute the focal enterprise. This figure also shows how material flows (MTS products and MTO products) move through the network.

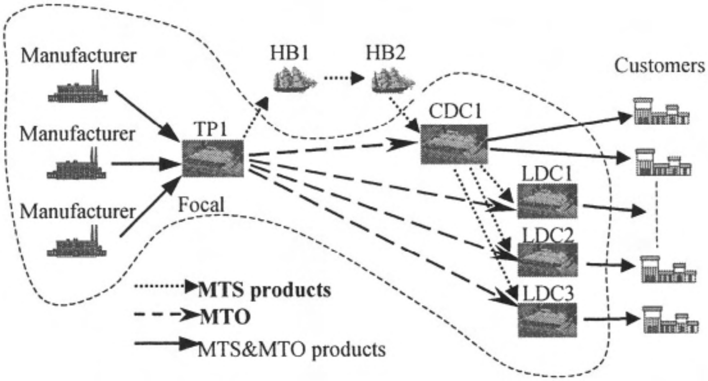


Figure 16-2. Production distribution network model

More specifically, current distribution network of the studied automotive enterprise is simulated as the reference scenario. It is a typical decentralized distribution solution, where one Consolidated Distribution Center (CDC) close to a harbor keeps strategic stock of MTS products and several local DCs act as intermediates to connect CDC with customers. As a result the responsiveness and customer service level are kept at an acceptable level. But keeping temporary stock at local DCs usually incurs high inventory costs. Thus, the company intends to investigate the possibility to re-configure their distribution network. The alternative is a centralized distribution solution that aims at reducing inventory costs by keeping all stocks at CDC. Thus the marginal inventory costs can be reduced very effectively. Whereas the lead time would not vary much since the CDC just covers a medium-size region.

Concerning the information flow, sixteen customer zones generate independent demands. The amount of demands follows a normal distribution and the inter-arrival time follows a Poisson distribution. MTS demand and MTO demand are separated according to the standard ratio (60% MTS and 40% MTO for each customer zone). Then all the demands are collected respectively by 4 DCs, consists of one consolidated DC (CDC1) and 3 local DCs (LDC1, LDC2 and LDC3). Subsequently MTO orders are sent directly to three manufacturers where the orders will be queued. On the contrary, MTS demands from LDC1, LDC2 and LDC3 are sent to CDC1. Based on the on-hand inventory level of CDC1, corresponding MTS cars are dispatched to local DCs where cars are finally delivered to customers.

In the consolidated DC, the inventory position of MTS products is reviewed periodically. A dynamic inventory policy $(s, \tilde{S})$ is employed to control strategic inventory, *"s"* is the given reorder point as usual. The order-up-to level "$\tilde{S}$" is determined by calculating the quantity needed between the time the order is placed and the time that the next period's order is received. In particular, "$\tilde{S}$" is unfixed and associated with demand forecast. For example, the inventory position *I(t)* is reviewed at time *t*. If *I(t)* is less than *s,* an order will be placed to replenish the inventory. The target is to cover the demand during period $(t, t+\Delta t)$, namely $\tilde{d}(t,t+\Delta t)$. So the order quantity *Q(t)* is determined by $Q(t) = s - I(t) + \tilde{d}(t,t+\Delta t)$. In this study, Δt is two month and $\tilde{d}(t,t+\Delta t)$ follows a given distribution representing the seasonality.

At the production stage, both MTO and MTS cars, are produced by three manufacturers regarding to given production ratio. Production preparation time is not considered separately but taken into account within the order handling time in this study. Finished cars are accumulated to a transit warehouse (TP1) by railroad. Then MTS cars will be moved to a harbor (HB1) by train, from where they will be transported by boat to another harbor (HB2) located among the customers. From HB2 they will be sent to CDC1 by train and subsequently dispatched to local DCs or delivered to customers. All the MTO cars are dispatched from TP1 to four DCs separately by train to meet their MTO demands. We assume that the handling capacity of each DC is unlimited. Thus all cars assigned to customers will be delivered at fixed time window consequently.

The production, transportation and dispatching processes of the centralized scenario are similar to the reference scenario (Figure 16-2), except that the centralized scenario contains just one consolidated DC to serve all customers.

4.2 Featured assumptions

For the sake of feasibility and simplicity, several assumptions concerning key distribution processes are introduced as following:

- According to a given standard ratio, customer demand is separated into two categories: demand for MTO products and demand for MTS products.
- Inventory position of MTS products in the consolidated DC is controlled based on a periodic review policy. When the net MTS inventory level is discovered to be less than reorder point, an order is placed to replenish the stock. The order-up-to level is dependant on the constant reorder point and dynamic demand forecast.
- MTS demand is filled from the strategic inventory of the consolidated DC. When demand exceeds the stock, unmet demand is backordered and delivered to customers as soon as it becomes available in stock. On the contrary, all MTO demands are backordered, and corresponding orders are sent to manufacturers immediately.
- Capacitated and continuous production is employed at manufacturers, where the order queue is processed in a First-In-First-Out (FIFO) manner.
- Transportation link is modeled as a finite capacity queue. The delivery frequency and batch size of each transportation link are given as input. At fixed time window, if the accumulated product amount can fill up a batch, a batch is sent out.
- When a MTS batch arrives at the consolidated DC, the queue of unmet MTS demands is checked. If the queue is empty, the entire batch is kept as a part of inventory. Otherwise, all the unmet MTS demands are to be filled consecutively until the arrival batch is emptied.
- In fixed time window, the queue of MTS and MTO products in DCs that are already assigned to specific customer is checked. And all ready products are delivered consequently. Loading time is included in the transportation time.

4.3 Simulation results analysis

According to these two scenarios of the case study, two simulation instances are extended, based on the simulation framework, to assist in performance comparison. Since configuration of distribution network is a typical strategic problem, simulation horizon is defined as two years to get long-term evaluation. Several KPIs are introduced, including average lead time, ratio of on-time delivery and average inventory level of DC. Average production costs and plant utilization ratio are also evaluated. In this chapter, we focus on two KPIs: lead time and inventory level. Comparison between both scenarios is presented in the following graphs. Considering the stochastic facet of the model, we simulate each scenario for 100 replications. Moreover, all the KPIs are calculated on the average.

Figure 16-3 shows the impact of reorder point upon lead time. "Avg-MTS" and "Avg- MTO" are average lead time for MTS and MTO products respectively. "Avg-Total" is average lead time for both categories.

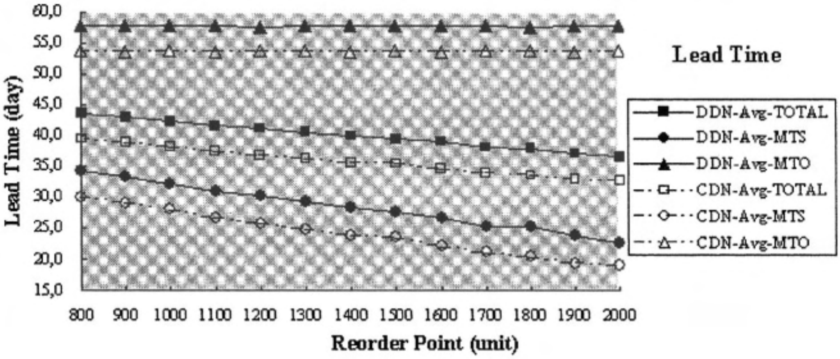


Figure 16-3. Comparison of lead time

For MTS products, average lead time of both scenarios will be reduced effectively when the reorder point goes up. On the contrary, lead time of MTO product is relatively stable. One unusual conclusion is that the total average lead time of CDN is less than that of DDN. Normally, the advantage of DDN is that more local DCs shorten delivery time between DC stage and customer stage. While in this case, the delivery time is negligible due to the limited service area. On the contrary, there are three transportation links between CDC1 and local DCs in DDN. In such case, extra waiting time is introduced in order to form necessary batch size (by train) for transportation. This is the very motivation to study centralized distribution network that covers just a middle-scaled area.

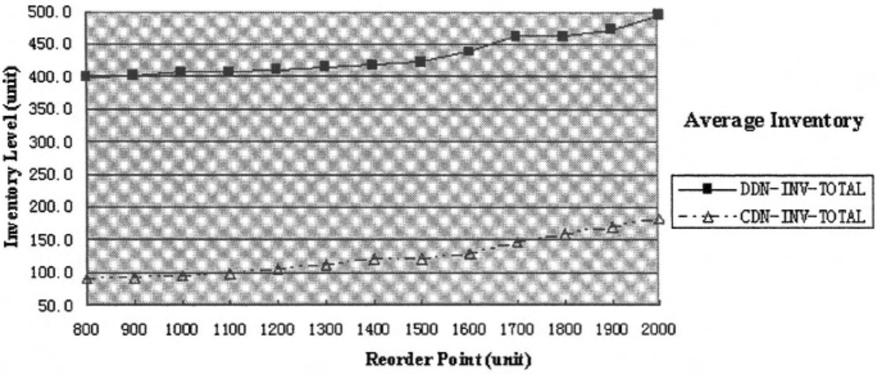


Figure 16-4. Comparison of average inventory level

Figure 16-4 shows average inventory level of both scenarios, where "INV-TOTAL" represents the total average inventory level for DDN and CDN respectively. In the decentralized scenario, MTS products demanded by local DCs have to wait in CDC1 for a period to form the batch. As a result, the average inventory level of DDN is higher in comparison with that of CDN. Obviously, the centralized solution will also achieve economy of scale by keeping all the stocks together. So corresponding inventory costs of DDN will be even higher than that of CDN.

Both simulation models are built in collaboration with logistic managers. The simulation results of DDN scenario are considered by logistic mangers as consistent with respect to actual operational data of the company.

5. CONCLUSION

A general modeling and simulation framework for supply chain network design is proposed. A number of facilities, constituents of supply chain network, are introduced as basic building blocks of the modeling and simulation package. These building blocks enable end users to build supply chain model quickly, simulate it with different operation rules and precisely evaluate its performances. An automotive case with both Make-to-Order and Make-to-Stock products is studied by way of scenario analysis. This multi-stage distribution network model is built following the modeling framework. Corresponding discrete event simulation model is developed by extending building blocks of the simulation package. More specifically, a decentralized distribution network of an automotive enterprise is simulated as the reference scenario and an alternative centralized solution is proposed and evaluated.

Concerning future research work, more attention should be paid to build general operation rules within the modeling and simulation framework. The possibility to combine the framework with specific optimization algorithm is also taken into account. Thus simulation optimization methodology can be employed to facilitate supply chain network design and other decisions related to SCM.

ACKNOWLEGEMENTS

This research is partially supported by the European Community research program, through the project ONE (Optimization methodology of Networked Enterprises, Project No. GRD1-2000-25710). These supports are gratefully acknowledged.

REFERENCES

1. Bhaskaran S. 1998. Simulation Analysis of a Manufacturing Supply Chain. *Decision Sciences,* 29, 633-657.
2. Breitman R.L., Lucas J.M. 1987. PLANETS: A Modeling System for Business Planning. *Interfaces,* 17, 94-106.
3. Cohen M.A., Lee H.L. 1988. Strategic Analysis of Integrated Production-Distribution Systems: Models and Methods. *Operational Research,* 36, 216-228.
4. Cohen M.A., Moon S. 1990. Impact of Production Scale Economies, Manufacturing Complexity and Transportation Costs on Supply Chain Facility Networks. *Journal of Manufacturing Operational Management,* 3, 269-292.
5. Geoffrion A.M., Powers R.F. 1995. Twenty Years of Strategic Distribution System Design: An Evolutionary Perspective. *Interfaces,* 25, 105-127.
6. Jansen D.R., Weert A., Beulens A.J.M., Huirne R.B.M. 2001. Simulation Model of Multi-Compartment Distribution in the Catering Supply Chain. *European Journal of Operational Research,* 133, 210-224.
7. Petrovic D., Roy R., Petrovic R. 1998. Modelling and Simulation of A Supply Chain in An Uncertain Environment. *European Journal of Operational Research,* 109, 299-309.
8. Petrovic D., Roy R., Petrovic R. 1999. Supply Chain Modelling with Fuzzy Sets. *International Journal of Production Economics,* 59, 443-453.
9. Simchi-Levi D., Kaminsky P., Simchi-Levi E. 2003. *Design and Management the Supply Chain: Concepts, Strategies and Case Studies.* 2nd Edition,McGraw-Hill/Irwin.
10. Slats P.A., Bhola B., Evers J., Dijkhuizen G. 1995. Logistic Chain Modeling. *European Journal of Operational Research,* 87, 1 -20.
11. Stock G.N., Greis N.P., Kasarda J.D. 1998. Logistics, Strategy and Structure: A Conceptual Framework. *International Journal of Operations and Production Management,* 18, 37-52.
12. Stevens G.C. 1989. Integration of the Supply Chain. *International Journal of Physical Distribution and Logistics Management,* 19, 3-8.

Chapter 17

INTERNET WEB-BASED INTEGRATION OF PROCESS AND MANUFACTURING RESOURCES PLANNING

Algirdas Bargelis, Rasa Mankutė

Abstract: This research deals with the integrated process and manufacturing resources planning system on the Web site. The algorithm and software of optimisation of processes and manufacturing resources planning for each alternative product and process are developed and presented. The appropriate technique for prediction the manufacturing resources at the early design stage of product is also presented. The obtained results could be placed on the Web site for two-way communication between customer and producer

Key words: manufacturing resources, process planning, Internet based technique.

1. INTRODUCTION

During the last few years, the manufacturing environment has greatly changed; it became a powerful and competitive weapon for mastering new methods of production. The need for designing new products and their manufacturing engineering has simultaneously increased. The new manufacturing strategy has been developed, that facilitates flexibility, reduces the design cycle time and the time of new products to the market [10, 12]. It is very significant for mechanical components because of their complexity and the most efficient and effective way to achieve the above-mentioned problems is to solve them by computer integrated manufacturing (CIM) systems; manufacturing engineering is one of CIM parts. The main objective of manufacturing engineering in the modern production environment is the integration of a process and manufacturing resources, finding the cheapest alternatives to produce new products. An urgent problem for many industrialized nations is the hollowing of a manufacturing section. The attractive low-cost manufacturing opportunities in the Far East, Eastern Europe and Latin America have lured many Western companies to subcontract the substantial parts of their existing processes with replacement their production division in the above-mentioned countries.

There are four main topics related to automation of manufacturing engineering of a new product. The first one is a set of developed computer-aided process planning (CAPP) systems [1, 9, 15, 16]. The second is the creation of manufacturing resources planning (MRP) systems [8, 14]. The third is the development of interfaces amongst CAPP and MRP, and the last fourth topic is the creation of the Web tool portal for collaboration among various customers and producers that are located in different countries and companies. The framework proposed can be used when customers make orders for manufacturing components or parts and customers assemble the products themselves. The aim of this framework is to raise the efficiency of collaboration of customers and producers aiming to maintain a business continuity management (BCM) in Global Manufacturing (GM) environment. Customers and manufacturers have different tasks: customers want to find the cheapest producers, while producers – to run their business.

The essential aim of our investigation was to develop and to generalise the theoretical methods of automated manufacturing engineering and the appropriate software that could realise the integrated approach of manufacturing engineering and product concept formation up to the component production, as well as to implement the obtained results in industrial activities aiming to increase both the efficiency and quality of the work. To attain this aim it was necessary not only to develop new methods and algorithms of both computer aided process and manufacturing resources planning but also to systematise and theoretically generalise the contradicting solutions, namely, Design for Manufacturing (DFM) and Design for Assembling (DFA) that were inevitable in creating new subsystems and assuring their integrity.

Based on our theoretical and experimental investigations the model of new product manufacturing engineering has been formulated covering both the product's concept design and its batch production stages. The new original method has been developed taking into account peculiarities of the manufacturing engineering and forecasting the production cost at the early stage of product design. For this purpose new methods of automated synthesis of manufacturing processes, material resources and machining time computation as well as the software have been created. Original interfaces between programmable modules and separate subsystems have been developed to ensure normal functioning of a complex manufacturing engineering model.

The main theoretical results of the investigation can be formulated as follows: scientific principles of rational manufacturing engineering of new products in the early stage of their design are created; these principles account for the peculiarities of modern production environment including different technological traditions and variety of products, and form up an integrated whole for new product design by means of manufacturing engineering and analysis systems.

Because of performed investigations, the methods of manufacturing process design and production resources planning are proposed to attain the level of a rational algorithm's and software. Invariant programmable modules are made up which consist of a variety of machining, assembling, painting and galvanising automated manufacturing engineering subsystems. These subsystems are being implemented in various mechanical and electrical engineering factories noticeably to speed up (3-5 times) new products manufacturing engineering and to improve the production quality. They enable to create and to analyse several variants of the manufacturing process in a short time, while changing materials and product design, and to optimise the solution. These systems have been continuously improved and the number of their functions has increased by applying the closed loop control.

All our subsystems can be connected to the Computer Integrated Manufacturing system, which will reduce the time and the budget of design and manufacturing engineering of new products. The research novelty is arrangement of the Web pages for specialized design of separate classes of products and their manufacturing processes including the development of optimal interfaces between the standard software.

2. THE OPTIMISATION OF MANUFACTURING ENGINEERING FOR MECHANICAL COMPONENTS

As it has been mentioned above, many researchers from various countries are developing the theory of automation manufacturing engineering, new approaches for integrated product development, etc. The basic role in achieving these aims belongs to computers and modelling of the processes and methods. Direct and indirect schemes can be used to categorise the role of computers in manufacturing. Our research field includes an indirect scheme of computers use i.e. the development of Computer Aided Process Planning (CAPP), Manufacturing Resources Planning (MRP II) subsystems operating in CIM, and modelling the optimal structure of automation of manufacturing engineering techniques. The simulation models for an early stage of the product design and a new approach of developing the automation theory of manufacturing engineering, particularly creating the functions of the production resources planning have been worked out [2-4]. The early stage of the new product design having the greatest influence on its quality, manufacturing costs and parameters of the product life cycle, the integrated system of automation of manufacturing engineering including the concept design stage of mechanical components, has been developed (Figure 17-1). The structure of the system is hierarchical and consists of two levels. The first level embraces the major elements of an early design stage of mechanical components and their processes. If the new product versions with their qualitative and quantitative parameters meet the market requirements, the product design is recommended and the second step of manufacturing engineering is started; if not – a new version of a product or its manufacturing engineering is generated. Such approach is available when mathematical modelling of the process in the conceptual design phase of a product or its component is created. During the modelling procedure all possible solutions of component manufacturing are to be analysed (Figure 17-1).

According to the modelling results, the optimisation of component design and its manufacturing is provided. The applied objective function of component costs C minimization is as follows (where M is the cost of component material; O is the cost of process operation; H is the overhead, and n is the number of operations):

$$C = M + \sum_{i=1}^{n} O_i + H. \qquad (1)$$

Some variants of component design and processes are possible. Many variants V with the consistently decreasing cost of component manufacturing $C(V_1) > C(V_2) >... > C(V_k) >... > C(V_m)$ have been created. The best variant offers the smallest cost. The second level of the system deals with the automation of the manufacturing engineering of mechanical components for batch production. To minimise and evaluate manufacturing costs in the concept design stage of a new mechanical component the technical-economic model for concurrent engineering environment has been created [3]. It consists of the following basic items:

1. Classification of products and calculation of material requirements;
2. Prediction of process planning and calculation of total and operational machining time;
3. Calculation of manufacturing costs and the rate of the machinery work efficiency.

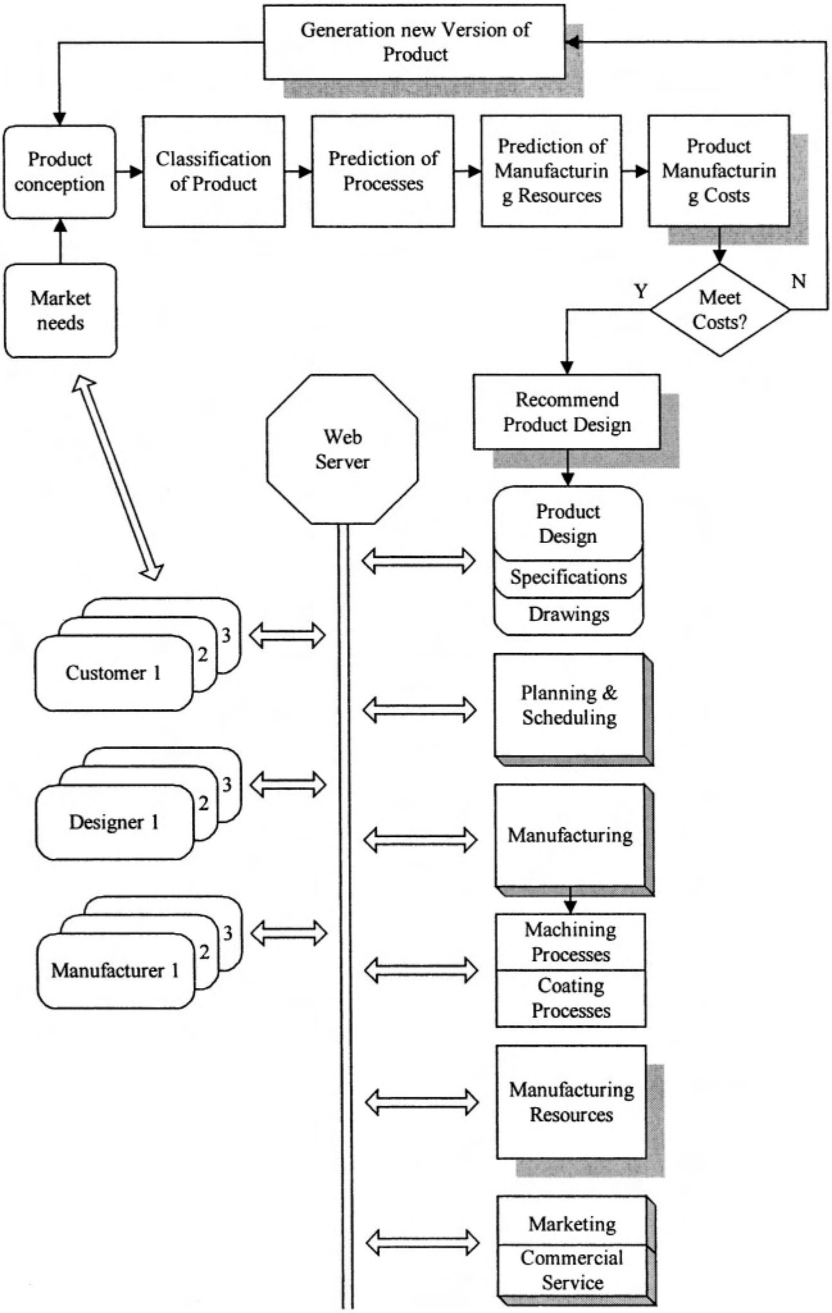


Figure 17-1. The structure of automation the manufacturing engineering system

All these parts can act either in the autonomous regime or in the integrated form using interfaces that ensure the information sharing among all users. In order to facilitate and accelerate the manufacturing engineering typical design features of components have been classified. A lot of them are divided into two classes – rotational and non-rotational. About 50 typical design features have been used for computer process planning, manufacturing resources planning and evaluation of manufacturing costs [3].

However, our simulation product justifies itself only when acting in the limited indeterminate space. To decrease the space of indetermination various means can be used. We have selected the classification of new mechanical components into separate class levels according to their quantitative and qualitative parameters. The components intended for the same class should have minimum scatter of these parameters.

Because of our investigation [4] data, several different classes of products made up of uniform typical design features (DF) have been formed. Different components despite they are made up of DF are characterised by different quantitative and qualitative parameters. Supposing there is a limited lot of classified objects Q=1, 2, 3, ..., m, where m is the object number, and there is a lot of attributes A=1, 2, 3, ..., n, characterising these objects, where n is the attribute number. Then a lot Q may be divided into p classes,

$$S_1, S_2, ..., S_p : Q = S_1 \cup S_2 ... \cup S_k ... \cup S_p . \qquad (2)$$

Two methods may be used to characterise object k by l attributes:

- <u>The first:</u> Quantitative attributes such as object mass, size, power, speed, etc. may be used; that means designation b_{kl} (k is the object number, l is the quantitative attribute) may be used.
- <u>The second:</u> It is possible to affirm that b_{kl} =1, if object k satisfies l attributes, and b_{kl} =0, if it does not satisfy them.

Any class of products S_k differs from the other one by the number of attributes or by their characteristics. Class S_k is defined by: the criterion of closeness of adjacent S_k; the criterion of exclusiveness and uniformity of adjacent $S_{k;}$; the generalised criterion, characterising the level of exclusiveness and uniformity of all classes. The presented classification method enables us to do optimal distribution of new unknown products to classes S_k. The algorithm and the program of unknown product distribution to some of classes S_k are realised. This analytical classification method of mechanical components can be applied in development of the future intelligent manufacturing systems.

3. THE MODELLING OF MANUFACTURING COSTS AT THE EARLY STAGE OF COMPONENT DESIGN

The next step of the classification of mechanical components is a calculation of the need of materials, their amount and types required for the production of the product. We have used a mathematical model for linear regression to calculate the requirement of various materials for the products of the same class S_{k}.. This is a comparatively simple model providing sufficient accuracy of calculation and is easily programmable. Other scientists [11] confirm the advantages of the model. However, the use of linear regression demands many statistical data. For calculation of different types and profiles of required metals and plastics, a linear regression equation of one type has been used. Regression coefficients were different. Qualitative and quantitative parameters of the product, its defined class (application, functions, mass, size, volume, density, etc.) were used as regressors.

Modelling the calculation of the need of material resources at an early stage of product design is based on prediction. Metals, plastics and chemicals are used in machine and apparatus building industry. Two models are being worked out for prediction and calculation of the materials need: the first one is applied for metals and plastics (nominal lot of materials M_i, i=1,...,n); the second – for chemicals; nominal lot of chemicals Ch_j, j=1, ..., m has been used. In both models, the forecast is based on the retrospective analysis. Materials are used in production applying traditional methods of all products (P), variation of the intensity of material streams in time (or when P design is changed) being monotonous.

To work out a retrospective model of the first type, functional relationships between material type and parameters (mass, volume, size, number of parts etc.) of any product are needed. Functional dependence of the products mass on cold rolled ferrous metal mass is presented. Such dependencies have to be derived for every product's parameter, having an effect on the materials need.

The calculation results of the need of material resources for commercial production of various S_k class products have been statistically analysed estimating five years data. These data were used for material resources prediction in the new product design. The forecast equation is based on a mathematical model of linear regression (where $\vec{Y}$ is the vector of predicted parameters; X is the regression coefficient matrix; $\vec{a}$ are the regressors and $\vec{e}$ are the vectors of errors):

$$\vec{Y} = X \cdot \vec{a} + \vec{e} \,. \tag{3}$$

These equations and the curves approximating them may be suitable for plastic or metal component. The need of material for coating processes is predicted according to the area of a coating surface.

In any case, the variation between the predicted and the calculation data of commercial production materials resources was less than 10-30%. The proposed simulation model for prediction of material requirements is used in Lithuanian machine and apparatus building industry.

The second stage of the model was devoted to prediction of processes and operations. The applied retrospective analysis of different production processes and operations has shown that the sort of material, the profile and the amounts consumed play a decisive role in the process route. Technological operations transform the form, size, and properties of the material or perform the assembly of separate parts. Different means of production or different technological operations, e.g., preparation, punching, welding, turning, milling, drilling, coating, assembling, which result in material transformation, are characterised by the form of initial materials and technological principles, such as plastic deformation, joining or disjoining of the components and painting. Any of these technological principles may be divided into technological operations and the order of priority of their performance may be assigned, e.g., the technological process of the product manufacturing may be formed. In this way any material requires original technological operations, e.g. if a product is manufactured by using thin sheet metal, (thickness $s \leq 3$ mm) the operations will be as follows: preparation, punching, bending, welding, drilling, screwing and coating. When the thickness of the sheet metal is increased, some of these operations are omitted, and instead the additional operations (milling, heat treatment, grinding) are introduced and all of them may be in one complete set (maximum) or may have various combinations.

The number of parts of the product, their size as well as the number of design features and their qualitative and quantitative parameters predetermine the combination of technological operations not only for metal sheets, but also for any type and profile of material (bars, angles or plastics).

Referring to the conventional processing methods the production process should be divided into preparation, flexible manufacturing systems (FMS) and a finishing stage. Preparation is used to form the processing bases of parts. FMS is used for the main production stage of the part, and finishing operations aim at manufacturing high quality parts or their design features.

The third stage of the model calculates the total and operational working hours of the product separately for preparatory, FMS and finishing processes.

Calculation of the operational working hours of the preparatory operations. We will present the calculation of labour consumption for the blanks of metal sheets, bars, angles, etc. at an early stage of component design [5, 6]. For the formation of the base surfaces and cutting of blanks, we have used statistical data obtained from commercial production computer aided process planning (CAPP) systems of practical application. The relationship of quantity M_j of predicted sheet metal (M_{NS}) or bar (M_{SK}) for product P_i of any class S_k and the quantity of rotational parts (N_{SK}) and non-rotational parts (N_{NS}) of a possible product is as follows (where c_1, c_2, m_{01}, m_{02} are experimental constants of function):

$$N_{SK} = c_1 \cdot (M_{SK})^{m_{01}}, \quad N_{NS} = c_2 \cdot (M_{NS})^{m_{02}}. \tag{4}$$

Preparation working hours and the quantity of parts machined are derived from experimental dependencies (working hours-removed volume of metal). Thus, the machine time of blanks cutting and datum surface preparation can be calculated in the form of this relationship:

$$T_{pSK} = T_{SK} + m_1 \cdot N_{SK}, \quad T_{pNS} = T_{NS} + m_2 \cdot N_{NS}. \tag{5}$$

Mobile or main manufacture of product parts is carried out on FMS. These processes depend on the type of part surfaces and are divided into two classes: 1) cylindrical (conical) exposed (CEDF) or unexposed (CNDF) for manufacturing and 2) flat exposed (FEDF) or unexposed (FNDF) for manufacturing design features. Machining time dependencies of a turning process and those of a milling process has been used [7]. Both these dependencies has been developed for fabrication processes of carbon steel of standard quality (surface roughness Rz 40) and are presented in Figure 17-2 and Figure 17-3.

The model is needed that enables the time calculation of product fabrication T without drawings and specifications at the early stage of product design. The abstraction of such model could be expressed (where S is the class level of a product, D is the geometrical form of a part, M is the type of material, QD are quantitative parameters of a part, QDF are qualitative parameters of design features, R is manufacturing traditions of an appropriate manufacturing system):

$$T \rightarrow f(S, D, QD, QDF, M, R). \tag{6}$$

It is possible to form various types of products, from their parts or the parts produced from typical design features. Undoubtedly, every different type of a product will sufficiently influence its manufacturing working hours. Applying the decision support theory and statistical data, function (6) may be expressed (where β is the coefficient evaluating the quality of machining process; k_m is the coefficient evaluating the material being machined, e.g. for high carbon steel k_m = 1,3):

$$T_{\min} = \sum_{i=1}^{n} T(CEDF)_i \cdot k_m \cdot \beta, \quad T_{\max} = \sum_{j=1}^{m} T(FNDF)_j \cdot k_m \cdot \beta. \tag{7}$$

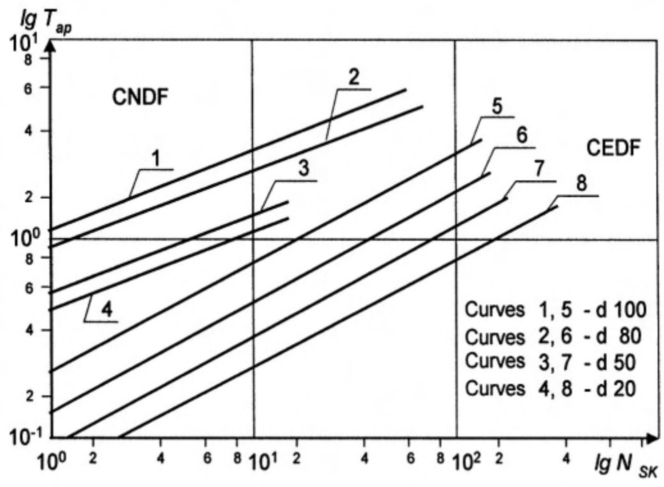


Figure 17-2. T prediction dependencies for rotational form of the work piece (d – diameter of bar)

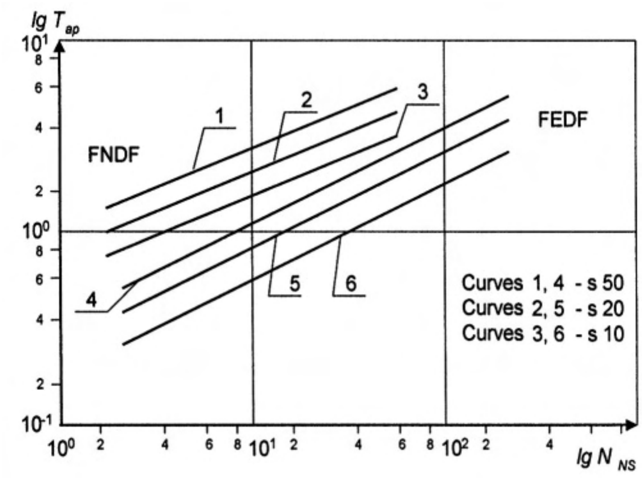


Figure 17-3. T prediction dependencies for non-rotational form of the work piece (s – thickness of sheet metal)

The equations (7) mean that the product is made only of CEDF or only of FNDF. The problem is solved according to the theory of chances and accepting that the plenty of design features of the product are dispersed evenly [5, 7].

4. IMPLEMENTATION OF DEVELOPMENT ON THE WEB SITE

Information processing is essential in manufacturing where products are produced to the base of available information from various sources. In recent years, Internet has become the worldwide information platform for sharing all kinds of information, and Intranet, which is based on Internet technology but is used within an organisation, has become a preferred platform for data sharing [13]. This approach provides a new role of Intranet/Internet technology for manufacturing, particularly for the companies struggling to gain a competitive edge in the global manufacturing area. Today's competitive manufacturing environment is illustrated as follows:

- Global competition;
- National characteristics of separate producers;
- Local competitive environment;
- Internal factors;
- Global marketplace.

Managers in the past were able to focus primarily on factors in the internal environment and the local national environment while determining their competitive stance. The managers of today must also identify and take into consideration competitive advantages associated with their nation of origin. Companies have to recognise all five of the above-mentioned levels if they wish to achieve and maintain a competitive position in the global and national marketplace.

We have proposed the Internet/Intranet based consumer-producer server architecture on the Web site (Figure 17-4) and have developed the software of the integrated process and manufacturing resources planning, which should help to solve the above-mentioned problems. The framework of a Web-based system is presented in Figure 17-5. This framework is of interest to the industry of Lithuania where so many small and medium enterprises (SMEs) are located, which produce the products that are developed in West Europe and the USA. The main idea of the Web-based system is that the customers searching cheapest producer can send their orders to several manufacturers anywhere.

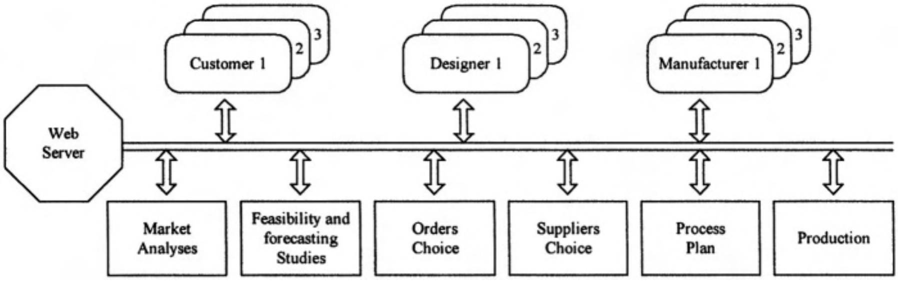


Figure 17-4. Customer-producer Web server architecture

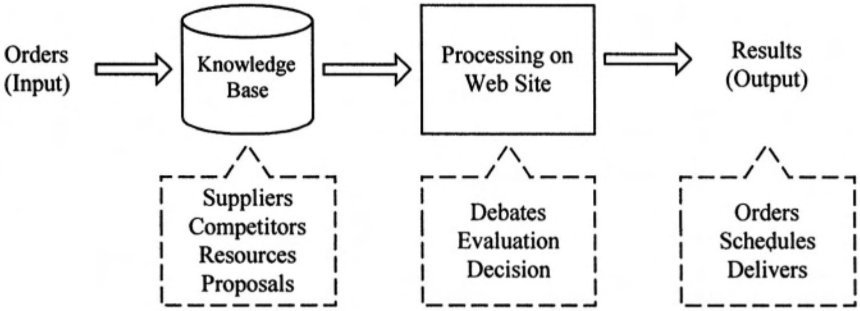


Figure 17-5. Framework of a Web-based integrated process and manufacturing resources planning system

5. CONCLUSIONS AND FINAL REMARKS

The main features of this research are the investigation of automation of mechanical components of manufacturing engineering using the same methods in both the batch production and the early stage of the product design. The systematic theoretically well-grounded forecasting method for machining and coating processes has been created. It includes the calculation of production resources at an early design stage of component. The investigation has resulted in the following developments:

1. A new multistage concept of manufacturing engineering has been proposed. It is based on the consequent data acquisition of a mechanical component at an early design stage and its batch production ensuring the automated synthesis of practically all products manufacturing processes accounting for various production traditions.

2. The model has been created for inertial prognosis of material resources. It predicts the requirements of a basic material for new mechanical components according to their functional, qualitative and quantitative parameters with 12 ... 33 percent accuracy.
3. The equation has been derived for predicting the total and operational working time needed for preparation and mechanical fabrication. The working time of mechanical components can be calculated in the early stage of the component design by means of this equation with 10 ... 37 percent accuracy.
4. The simulation model of prediction of a new product manufacturing cost at the early stage of its design has been developed that makes it possible to take an advance in checking product compatibility on the market and to modify its quantitative-qualitative parameters and its manufacturing processes.
5. The role of Intranet/Internet technology in manufacturing is emphasised. The presented architecture of customer – producer server on the Web site could help in fast evaluation the company's feasibility and manufacturing costs of the current order. The main subunit of this development is software of an integrated process and manufacturing resources planning.

ACKNOWLEDGEMENTS

This research was supported by the EU Tempus Jep project No: AC_JEP_13183-98 "Education for Integrated Design and Manufacturing" 1999-2001 year.

REFERENCES

1. Atkinson A.C., 1994. Fast very robust methods for the detection of multiple outliers. In: *JASA* 89, 1329–1339.
2. Badawy M.K., 1999. Technology Management Education: Alternative Modules. *IEEE Engineering Management Review,* 27(2), 55–67.
3. Bargelis A., 1995. Modelling to Predict Material Resources for Manufacturing New Products. In: *Proceedings XIV International Conference IASTED, Modelling, Identification and Control,* Igls, Austria, 20-22 February, 1995, 93–94.
4. Bargelis A., 1996. Developing of Technical-Economic Modules for Concurrent Engineering Environment. In: *Proceedings of the 12th Int. Conf. CAD / CAM Robotics and Factories of the Future,* London, 14-16 August, 1996, 902–907.
5. Bargelis A., 1996. Estimation and Minimisation of Costs for Experimental Design Products and Processes. In: *Proceedings of the 41st. Int. Scientific Colloquium,* Ilmenau: Technical University, 23-26 September, 1996, Part 1, 529–534.

6. Bargelis A., 1996. The Modelling Costs of Manufacturing New Products at the Early Stage of Its Design. In: *Proceedings of the Fifteenth IASTED International Conference,* Insbruck, Austria, 19 - 21 February, 1996, 265–267.
7. Bargelis A., 1997. The Modelling of Production Time in Manufacturing Engineering. *Mechanika.* Kaunas: Technologija, ISSN 1392-1207, 4(11), 50–54.
8. Bonsack R., 1986. Cost Accouting in the Factory of the Future. *CIM Review,* 2(3), 28–32.
9. Eimaraghy H., 1993. Evolution and future perspectives of CAPP. *Annals of the CIRP,* 42(2), 1–13.
10. Iskandar B.Y., Kurokawa S., LeBlanc L.J., 2001. Adoption of Electronic Data Interchange: The Role of Buyer-Supplier Relationships. *Transactions on Engineering Management,* 4, 505–517.
11. Kelly P.F., Drury J.C., Weston W., Devine N., 1995. Multiple Linear Regression as a Basis for Cost Oriented Decision Support. In: *Proceedings of the Eleventh National Conference on Manufacturing Research,* London: Taylor & Francis, 579–586.
12. Kim W.Ch., Maugborne R., 2002. Charting your Company's Future. *Harvard Business Review,* June, 77–83.
13. Lau H., 1998. The New Role of Intranet/ Internet Technology for Manufacturing. *Eng. with computers,* 14, 150–155.
14. Lee-Mortimer A., 1990. The Manufacturing Engineer of the Future. *Integrated Manufacturing Systems,* July, 1990, p. 115.
15. Marri H.B., Gunasekaran A., Grieve R.J., 1998. Computer Aided Process Planning: A State of Art. *Int.Journal Advanced Manufacturing Technology,* 14, 261–268.
16. Tang Y.S., Gao J., Bennett G., 1996. Computer - aided Process Planning for Fabrication Applications. In: *Proceedings ofthe 12th Int. Conf. CAD/CAM, Robotics and Factories of the Future,* London, 1061–1066.

Chapter 18

VISUAL REPRESENTATION OF MATERIAL FLOWS IN CHEMICAL PLANT SYSTEMS

An approach based on SCHEDULE++

Nadezhda Sotskova, Jörg Haustein, Winfried Jänicke, Wolfgang Thämelt

Abstract: This chapter discusses the utilisation of software add-ons like SCHEDULE++ for the supply chain manager, production planner and material requirements planning controller. Using this tool, the flow of material can be tracked over several production and storage facilities of chemical plants. Moreover, supply chain tree and dynamic pegging data for material production and material consumption can be gathered. It will be shown, how consequences of changes in raw material delivery, changes to customer orders, or machine breakdowns can be visualised along the supply chain. A recursive algorithm was developed for dynamic pegging over the whole supply chain. We also discuss how such software applications can be used together with the SAP R/3 system.

Key words: supply chain, production planning, recursive algorithm.

1. INTRODUCTION

This chapter will describe production planning in a system of chemical plants using standard Production Planning (PP) systems. A planning situation in a PP system is characterised by plant stocks (factory inventory), planned orders, process orders, purchase orders, purchase requisitions, which are planned independently of customer orders. There may be several kinds of breakdowns for a planning situation. The following situations will be considered in this scenario:

1. A delivery of raw material is delayed (i.e., a purchase order or purchase requisition has to be changed).
2. There is a machine breakdown (i.e., process orders, which are assigned to the machine must be deleted or assigned to another machine).

3. One of several competing process orders on a bottleneck resource has to be selected by taking into account the effect on the whole supply chain.
4. A customer order has to be shifted (to an earlier or later date) or cancelled.

All of the above situations lead to the desire to see the effects on all the plants in the supply chain (SC). In the first case one would like to know, which process orders are affected and which customer orders cannot be delivered on time. In the second scenario (similar to the case 1), the breakdown leads to material unavailability, and the effects on subsequent steps have to be taken into account. The decision in the third example has to be made under consideration of the dependant customer orders, with respect to their delivery dates and strategic significance. In the fourth case, capacities can be freed and raw material purchase can be reduced. If all of these corollaries can not be considered, promises to customers cannot be kept or the stock levels will increase.

The chapter is organized as follows. In Section 2 the problem will be specified in detail. The solution approach is briefly described in Section 3. Section 4 contains descriptions of the main components of the SCHEDULE++ software add-on to SAP R/3. The algorithm for constructing a tree of the material flow in a system of plants is described in Section 5.

2. PROBLEM SPECIFICATION

Material requirements planning (MRP) or consumption-based planning can be used as the materials planning procedure for resolving the conflicts described above. MRP tries to optimise the following possibly competing goals: Optimising the service level, minimising costs and capital lockup. The main role of MRP is to monitor stocks, i.e., to generate order proposals for purchase and production (planned orders, purchase requisitions, or delivery schedules). Standard PP systems provide MRP runs, but sometimes an MRP run can result in invalid plans with capacity conflicts on resources. In addition, the requirements planning in chemical multi-purpose plants often consists of merging a large number of Microsoft Excel sheets, so that the whole current planning situation often remains unclear.

Planning that spans multiple plants may also be desirable. Unfortunately, planning processes in chemical multi-purpose plants are usually organised in the form of anonymous make-to-stock manufacturing. Therefore there are no data structures representing links in the supply chain. Moreover, these data structures are often not even wanted, because the idea behind anonymous make-to-stock manufacturing is precisely to allow product allocation alterations.

These problems need systems that can connect different planning situations at run time. This can be additional systems (e.g., Advanced Planning and Optimisation Systems) with their own server, database, data model, infrastructure, and integration model. The alternative to these solutions are genuine add-ons to well-known PP or ERP (Enterprise Resource Planning) systems.

3. PROPOSED APPROACH

In this section the solution to the above-mentioned problems will be described and exemplified.

3.1 Simultaneous material requirement and capacity planning (MRCP) using SCHEDULE++

The scheduling and information processing system SCHEDULE++ (developed by OR Soft Jänicke GmbH) can be used for detailed planning, requirements planning and supply chain co-ordination. SCHEDULE++ can be used as add-on to established standard ERP systems (such as SAP R/3 [3]) and as standalone system. Although SCHEDULE++ completely fulfils all requirements for planning system products, its philosophy differs in substantial points from other planning tools [2].

As software add-on SCHEDULE++ can be used to plan material availability, resource availability and capacity utilisation simultaneously. In that way the following planning requirements can be handled: Maintenance planning, personnel planning, sequence optimisation, handling of different lot sizes, etc.

SCHEDULE++ promotes a decentralised supply chain management (SCM) instead of the usual centralised SC co-ordination. Decentralised SCM is cross-linking of individual production departments with preservation of important autonomy conditions. With the help of SCM, local planning decisions can be made using global parameters.

3.2 Main idea

The main idea for the solution proposed here is to use the material stock/requirement lists in the existing PP system (MM-IM – Material Management and Inventory Management). For these material stock/requirement lists an algorithm can be developed that generates a dynamic tree of the material flow at run time.

The algorithm was developed for the two possible "directions" of such a tree. "Forward" looks for matching material requirements, and it will find the requirements in a supply chain that are met by a given material production or purchase. "Backward" tries to allocate production of materials or purchase along the supply chain according to a given material requirements. The dynamic supply chain tree can therefore start from a purchase, process or customer order, depending on the "direction".

Supply chain trees can be used to evaluate and highlight the following conflicts:

- Which process/customer orders are affected by a delay or cancellation of material delivery?
- Which process/customer orders are affected by the rescheduling of a process order?
- Which process/purchase orders can be deleted/reassigned if a customer order was cancelled?

It is also possible to generate a curve and a Gantt Chart for a specific material flow (dynamic view). These charts visualise the material stock/requirement lists and resource utilisation of all materials and resources involved in the production process of a certain finished product. They can also integrate information from multiple clients and PP systems.

By placing a copy of the production model (without changing PP system data) into the RAM (Random Access Memory) of a client PC, a number of algorithms, transformations, and graphical user interfaces can be provided, e.g.:

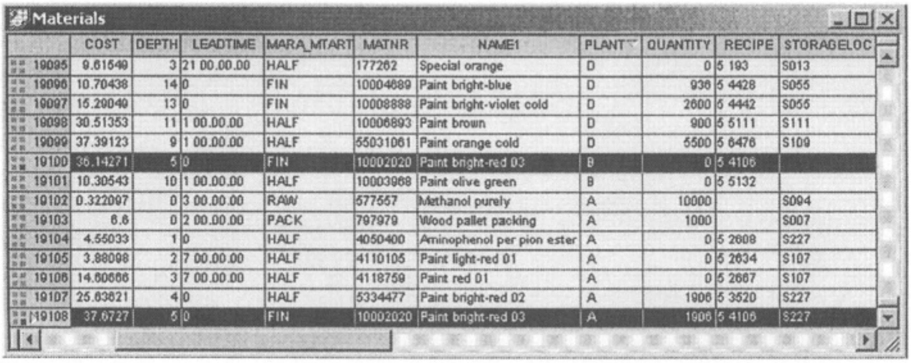
- Graphic visualisation of capacities of production units, equipment, personnel, stock as Gantt Charts or histograms.
- Representing the flow of material in histograms and trees.
- Displaying and summarising data objects, e.g., customer orders, process orders, planned orders, purchase orders in virtual browsers (work sheets).
- Constructing dynamic views of material flow for the supply chain and dynamic pegging of material production and consumption.
- Data from SCHEDULE++ browser can be easily mirrored to user-specified Microsoft Excel tables [1].

Using the SCHEDULE++ system, one can perform complex supply chain planning operations from forecasting through production to distribution and sales. It must be mentioned that after each change of the planning situation, sometimes a completely new allocation of process orders must be found. Therefore, it makes sense for planners, MRP controllers and SC managers to construct and analyse dynamic material flow trees on the basis of information about new customer orders and several kinds of breakdowns along with scheduling and planning. Such pegging views can be constructed in seconds for each material (half-finished and finished product) and usually for the flow of material a "First In, First Out" rule is presupposed. Moreover, decision makers can simulate changes to all data objects and thus improve a current plan before planning decisions are made. A satisfactory plan can be written back to the leading system immediately.

3.3 Illustrative example

In order to illustrate these trends a real chemical plant will be used as an example after the confidentiality of the data has been assured. Some field names and field contents will be in German, since the data were derived from a German enterprise.

The following figures are typically used in SCHEDULE++ for data display. Material master data is presented in a browser (Excel like structure) in Figure 18-1. Bill of material (BOM) information is presented as cascade in Figure 18-2. Each product has a master recipe, i.e., an exact description of a production technology. Recipes describe manufacturing processes and contain lists of materials (BOM) and lists of resources (manufacturing lines, processing units, machines, vessels, tanks, warehouse locations, workforce) necessary for the production of a given material. Materials without recipes are raw materials (RAW) or materials that can be produced in another plant. Finished products are indicated by FIN. Semi-finished products (HALF) and packaged product (PACK) are also sometimes sold.

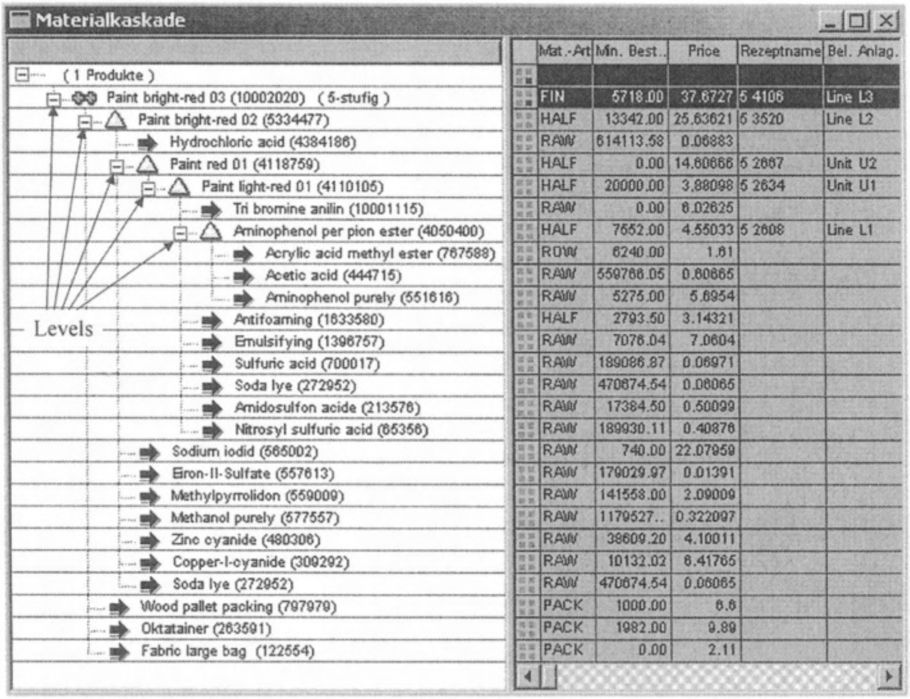


	COST	DEPTH	LEADTIME	MARA_MTART	MATNR	NAME1	PLANT	QUANTITY	RECIPE	STORAGELOC
19095	9.61549	3	21 00.00.00	HALF	177262	Special orange	D	0	5 193	S013
19096	10.70438	14	0	FIN	10004689	Paint bright-blue	D	936	5 4428	S055
19097	15.29049	13	0	FIN	10008888	Paint bright-violet cold	D	2600	5 4442	S055
19098	30.51353	11	1 00.00.00	HALF	10006893	Paint brown	D	900	5 5111	S111
19099	37.39123	9	1 00.00.00	HALF	55031061	Paint orange cold	D	5500	5 6476	S109
19100	36.14271	5	0	FIN	10002020	Paint bright-red 03	B	0	5 4106	
19101	10.30543	10	1 00.00.00	HALF	10003968	Paint olive green	B	0	5 5132	
19102	0.322097	0	3 00.00.00	RAW	577557	Methanol purely	A	10000		S094
19103	6.6	0	2 00.00.00	PACK	797979	Wood pallet packing	A	1000		S007
19104	4.55033	1	0	HALF	4050400	Aminophenol per pion ester	A	0	5 2608	S227
19105	3.88098	2	7 00.00.00	HALF	4110105	Paint light-red 01	A	0	5 2634	S107
19106	14.60666	3	7 00.00.00	HALF	4118759	Paint red 01	A	0	5 2667	S107
19107	25.63821	4	0	HALF	5334477	Paint bright-red 02	A	1906	5 3520	S227
19108	37.6727	5	0	FIN	10002020	Paint bright-red 03	A	1906	5 4106	S227

Figure 18-1. View of some materials

Let us consider one material in the presented material master and cascade view, namely the final product 10002020 (see lines labelled

19100 and 19108 in Figure 18-1). This material has the following properties: Name (paint bright-red 03), number (10002020), recipe (5 4106), production plants (A or B), quantity of customer order (e.g., 1906 kg on plant A), bill of material (cascade view in Figure 18-2), and depth of materials cascade (equal to 5). The BOM consists of five production levels (the top one is included). Four semi-finished materials are produced: Paint bright-red 02, paint red 01, paint light-red 01 and material 4050400 on the basis of recipes: 5 3520, 5 2667, 5 2634, 5 2608, respectively, which use various raw materials and packing products (see Figure 18-2).



	Mat.-Art	Min. Best..	Price	Rezeptname	Bel. Anlag.
(1 Produkte)					
Paint bright-red 03 (10002020) (5-stufig)	FIN	5718.00	37.6727	5 4106	Line L3
Paint bright-red 02 (5334477)	HALF	13342.00	25.63621	5 3520	Line L2
Hydrochloric acid (4384186)	RAW	614113.58	0.06883		
Paint red 01 (4118759)	HALF	0.00	14.60666	5 2667	Unit U2
Paint light-red 01 (4110105)	HALF	20000.00	3.88098	5 2634	Unit U1
Tri bromine anilin (10001115)	RAW	0.00	6.02625		
Aminophenol per pion ester (4050400)	HALF	7652.00	4.55033	5 2608	Line L1
Acrylic acid methyl ester (767588)	ROW	6240.00	1.61		
Acetic acid (444715)	RAW	559766.05	0.60665		
Aminophenol purely (551616)	RAW	5275.00	5.6954		
Antifoaming (1633580)	HALF	2793.50	3.14321		
Emulsifying (1396757)	RAW	7076.04	7.0604		
Sulfuric acid (700017)	RAW	189086.87	0.06971		
Soda lye (272952)	RAW	470674.54	0.06065		
Amidosulfon acide (213576)	RAW	17384.50	0.50099		
Nitrosyl sulfuric acid (65356)	RAW	189930.11	0.40876		
Sodium iodid (565002)	RAW	740.00	22.07959		
Eiron-II-Sulfate (557613)	RAW	179029.97	0.01391		
Methylpyrrolidon (559009)	RAW	141558.00	2.09009		
Methanol purely (577557)	RAW	1179527..	0.322097		
Zinc cyanide (480306)	RAW	38609.20	4.10011		
Copper-I-cyanide (309292)	RAW	10132.02	6.41765		
Soda lye (272952)	RAW	470674.54	0.06065		
Wood pallet packing (797979)	PACK	1000.00	6.6		
Oktatainer (263591)	PACK	1982.00	9.89		
Fabric large bag (122554)	PACK	0.00	2.11		

Figure 18-2. Cascade view of material 10002020

Next, let us consider the planning situation view. Schedules constructed in SCHEDULE++ can be presented in form of tables or as Gantt Charts. The whole detailed production cascade of the discussed material 10002020 for two months is presented in Figure 18-3 as Gantt Chart, which graphically shows the earliest and latest time of processing steps on resources (no other production shown). The materials stocks are represented as histograms (see Figure 18-3). A material supply or a process order step results in a continuous or instantaneous change of a material quantity in a histogram. The length of the displayed production interval (time axis) can be selected and modified by the user. Thus, it is possible to show the production situation for several hours, days or even years.

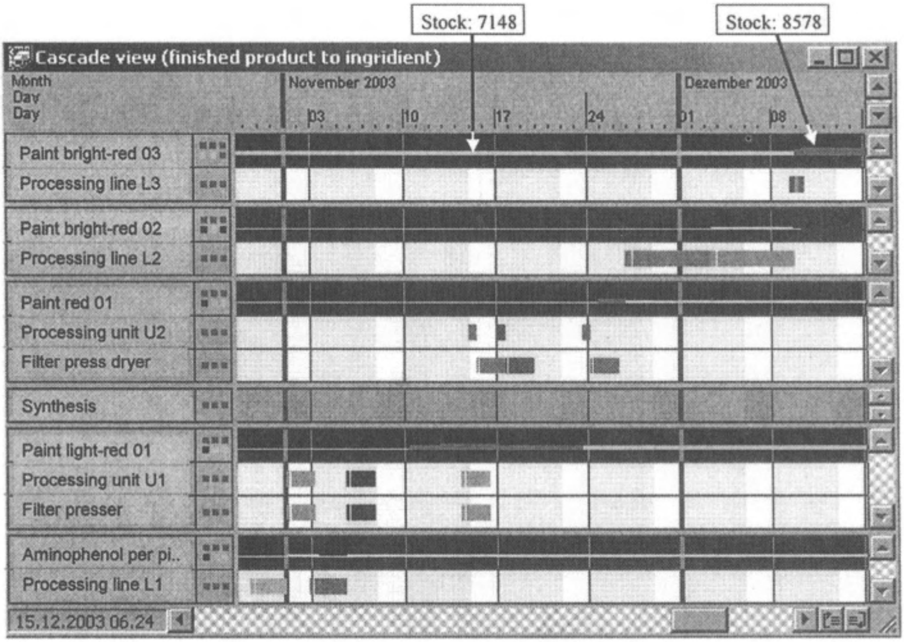
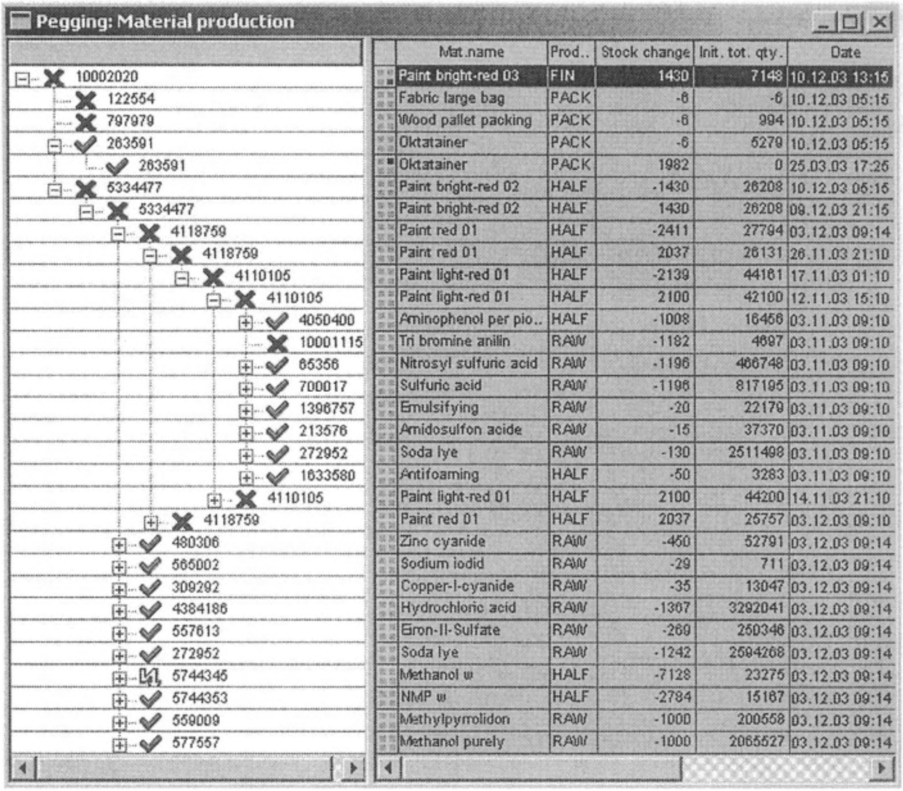


Figure 18-3. Planning situation view of final product 10002020

Finally, the dynamic supply chain trees can be helpful to see the production conflicts. Figure 18-4 shows production pegging in SCHEDULE++ for final material 10002020 ("forward" flow of material). Figure 18-5 demonstrates consumption pegging for semi-finished material 4050400 ("backward" flow of material). One can see that at the begin of the long-range planning period in November, 2003, there were 7148 kg of product 10002020 in the warehouse (see the first histogram for paint bright-red 03 in Figure 18-3 and/or the field "Initial total quantity" in Figure 18-4). Those 7148 kg were prepared and allocated for some customers. Out of this total stock, 1430 kg are produced by process orders using the different resources (processing line L1, filter presser, processing unit U1 and so on). Production can't be increased because there is no more raw materials available (e.g., raw material 10001115 and others). Look at Figure 18-4 use the symbols next to the material numbers to better understand the example (see Table 18-1).



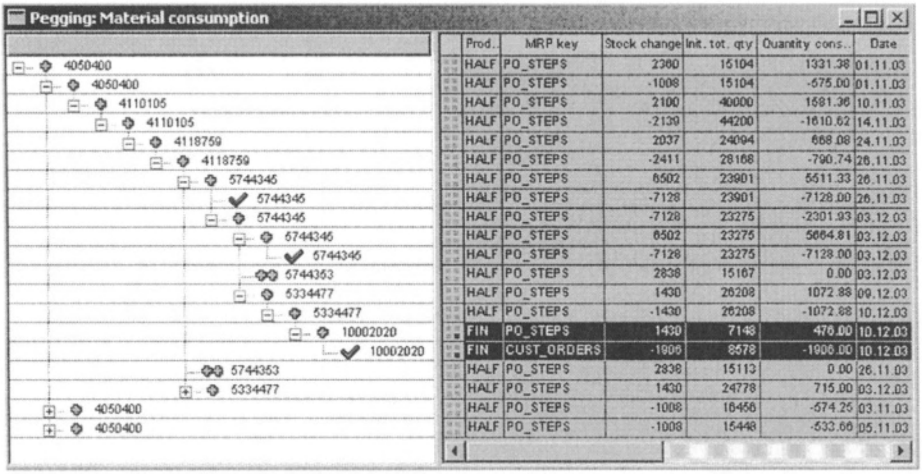
Pegging: Material production

```
⊟ ✗ 10002020
    ✗ 122554
    ✗ 797979
  ⊟ ✓ 263591
      ✓ 263591
  ⊟ ✗ 5334477
    ⊟ ✗ 5334477
      ⊟ ✗ 4118759
        ⊟ ✗ 4118759
          ⊟ ✗ 4110105
            ⊟ ✗ 4110105
              ⊞ ✓ 4050400
                ✗ 10001115
              ⊞ ✓ 65356
              ⊞ ✓ 700017
              ⊞ ✓ 1396757
              ⊞ ✓ 213576
              ⊞ ✓ 272952
              ⊞ ✓ 1633580
            ⊞ ✗ 4110105
        ⊞ ✗ 4118759
      ⊞ ✓ 480306
      ⊞ ✓ 565002
      ⊞ ✓ 309292
      ⊞ ✓ 4384186
      ⊞ ✓ 557613
      ⊞ ✓ 272952
      ⊞ ⚡ 5744345
      ⊞ ✓ 5744353
      ⊞ ✓ 559009
      ⊞ ✓ 577557
```

Mat.name	Prod..	Stock change	Init. tot. qty.	Date
Paint bright-red 03	FIN	1430	7148	10.12.03 13:15
Fabric large bag	PACK	-6	-6	10.12.03 05:15
Wood pallet packing	PACK	-6	994	10.12.03 05:15
Oktatainer	PACK	-6	5279	10.12.03 05:15
Oktatainer	PACK	1982	0	25.03.03 17:26
Paint bright-red 02	HALF	-1430	26208	10.12.03 05:15
Paint bright-red 02	HALF	1430	26208	09.12.03 21:15
Paint red 01	HALF	-2411	27794	03.12.03 09:14
Paint red 01	HALF	2037	26131	26.11.03 21:10
Paint light-red 01	HALF	-2139	44161	17.11.03 01:10
Paint light-red 01	HALF	2100	42100	12.11.03 15:10
Aminophenol per pio..	HALF	-1008	16456	03.11.03 09:10
Tri bromine anilin	RAW	-1182	4697	03.11.03 09:10
Nitrosyl sulfuric acid	RAW	-1196	466748	03.11.03 09:10
Sulfuric acid	RAW	-1196	817195	03.11.03 09:10
Emulsifying	RAW	-20	22179	03.11.03 09:10
Amidosulfon acide	RAW	-15	37370	03.11.03 09:10
Soda lye	RAW	-130	2511498	03.11.03 09:10
Antifoaming	HALF	-50	3283	03.11.03 09:10
Paint light-red 01	HALF	2100	44200	14.11.03 21:10
Paint red 01	HALF	2037	25757	03.12.03 09:10
Zinc cyanide	RAW	-450	52791	03.12.03 09:14
Sodium iodid	RAW	-29	711	03.12.03 09:14
Copper-I-cyanide	RAW	-35	13047	03.12.03 09:14
Hydrochloric acid	RAW	-1307	3292041	03.12.03 09:14
Eiron-II-Sulfate	RAW	-269	250346	03.12.03 09:14
Soda lye	RAW	-1242	2594268	03.12.03 09:14
Methanol w	HALF	-7128	23275	03.12.03 09:14
NMP w	HALF	-2784	15167	03.12.03 09:14
Methylpyrrolidon	RAW	-1000	200558	03.12.03 09:14
Methanol purely	RAW	-1000	2065527	03.12.03 09:14

Figure 18-4. Dynamic pegging for production of material 10002020

Since some customer order needs 1906 kg of material 10002020 (see the fields "Stock change"and "Quantity consumed" in Figure 18-5) on 10 December, 2003, the process orders achieve the fulfilment of the customer order to 75% (1430 kg / 1906 kg * 100%). It is obvious from Figure 18-4 and Figure 18-5 that the customer order can be satisfied only with 75 %, provided that the purchasing organisation will buy sufficient quantity of raw material 10001115 and that other conflicts will be resolved (see red crosses and yellow lightning in Figure 18-4). Thus, negotiations with the customer are required to reach a new delivery date for full 100 % production.



Pegging: Material consumption

	Prod.	MRP key	Stock change	Init. tot. qty	Quantity cons..	Date
⊟ 4050400	HALF	PO_STEPS	2380	15104	1331.38	01.11.03
⊟ 4050400	HALF	PO_STEPS	-1008	15104	-575.00	01.11.03
⊟ 4110105	HALF	PO_STEPS	2100	40000	1581.36	10.11.03
⊟ 4110105	HALF	PO_STEPS	-2139	44200	-1610.62	14.11.03
⊟ 4118759	HALF	PO_STEPS	2037	24094	668.08	24.11.03
⊟ 4118759	HALF	PO_STEPS	-2411	28168	-790.74	26.11.03
⊟ 5744345	HALF	PO_STEPS	6502	23901	5511.33	26.11.03
✓ 5744345	HALF	PO_STEPS	-7128	23901	-7128.00	26.11.03
⊟ 5744345	HALF	PO_STEPS	-7128	23275	-2301.93	03.12.03
⊟ 5744345	HALF	PO_STEPS	6502	23275	5664.81	03.12.03
✓ 5744345	HALF	PO_STEPS	-7128	23275	-7128.00	03.12.03
5744353	HALF	PO_STEPS	2838	15167	0.00	03.12.03
⊟ 5334477	HALF	PO_STEPS	1430	26208	1072.88	09.12.03
⊟ 5334477	HALF	PO_STEPS	-1430	26208	-1072.88	10.12.03
⊟ 10002020	FIN	PO_STEPS	1430	7148	476.00	10.12.03
✓ 10002020	FIN	CUST_ORDERS	-1906	8578	-1906.00	10.12.03
5744353	HALF	PO_STEPS	2838	15113	0.00	26.11.03
⊞ 5334477	HALF	PO_STEPS	1430	24778	715.00	03.12.03
⊞ 4050400	HALF	PO_STEPS	-1008	16456	-574.25	03.11.03
⊞ 4050400	HALF	PO_STEPS	-1008	15448	-533.66	05.11.03

Figure 18-5. Dynamic pegging for consumption of material 4050400

Table 18-1. Symbols used in figures above

✚✚	Double plus in Figure 18-2	Final product
△	Yellow triangle in Figure 18-2	Production phase for semi-finished product
➡	Black pointer in Figure 18-2	Raw material, semi-finished material or package
✓	Green hook in Figure 18-4	Order is covered completely by suppliers
✖	Red cross in Figure 18-4	Order is not covered at all by suppliers
⚡	Yellow lightning in Figure 18-4	Order is only partly covered by suppliers
✓	Green hook in Figure 18-5	Material of the order is used completely by pegging
✚	Plus in Figure 18-5	Material of the order is partly used by pegging (partly put in stock)
✚✚	Double plus in Figure 18-5	Order does not have a pegging
⊟		Level opened
⊞		Level closed

4. SCHEDULE++ DESCRIPTION

Next, we give a short description of the main components of the scheduling and information processing system SCHEDULE++.

4.1 SCHEDULE++ Add-on to SAP R/3

The SCHEDULE++ add-on to SAP R/3 downloads business data from the R/3 logistics components PP-PI (Production Planning in Process Industry) or PP, SD (Sales and Distribution) and MM-IM. SCHEDULE++ has a modular structure with the following main modules.

- The *kernel* which is responsible for the internal data handling, the coordination of all other scheduler modules, all algorithmic tasks (e.g., scheduling, pegging).
- The *graphical user interface.*
- *Data exchange modules.*

4.2 SCHEDULE++ data exchange module

The SCHEDULE++ data exchange modules utilise standard interface functionalities. Currently, SCHEDULE++ data can be read from and saved to the SAP R/3 system *(SAP add-on),* ORACLE database *(standalone version)* and other business, operations management and process control systems via special interfaces. The SCHEDULE++ data exchange module (for SAP R/3) transforms SAP R/3 business objects into SCHEDULE++ data objects and exchange handles data with the external system. It communicates with the SCHEDULE++ kernel that holds the data objects in the SCHEDULE++ *data dictionary* (internal SCHEDULE++ data organisation).

A single SCHEDULE++ data exchange module can handle multiple independent *SAP connections* at the same time and can therefore integrate business data from multiple SAP R/3 systems in a common SCHEDULE++ data dictionary.

4.3 Internal SCHEDULE++ data structures

SCHEDULE ++ (as *SAP add-on*) operates on a copy of the SAP R/3 master and dynamic data and uses no own planning data. Various software components collaborate with one another to provide access to the SAP R/3 system and to transform R/3 business model data to SCHEDULE++ business model data, which is stored in the SCHEDULE++ *data dictionary.* They are implemented as distributed services on the *client PC* (the PC that runs the SCHEDULE++ kernel and data exchange module) and the *SAP R/3 application server.*

Event lists are special highly efficient internal SCHEDULE++ data structures that contain time-dependent information about the planning and scheduling situations. Currently, different types of event lists are available for resources and materials. Events are created by dynamic data objects, i.e., by planned orders, purchase orders, customer orders, plant stocks, and process orders.

Event lists can be queried for various saved planning information by using *event list data fields.* For example, the event list data field “DEPENDANTS” yields the *dynamic pegging(s)* for a given data object.

Dynamic peggings are those data objects on the event list that are linked to the given object by a material requirement. The event list data field "DEPENDANTS" yields the rate (between 0 and 1) that it has in creating or fulfilling the material requirement for the given data object. For example, a process order may require 500 kg of a certain raw material, which are supplied by two purchase orders of 200 kg and 300 kg. The event list data field "DEPENDANTS" would then return those to purchase orders as dynamic peggings, with the fulfilment rates of0.4 and 0.6, respectively.

5. ALGORITHM FOR THE CONSTRUCTION OF THE MATERIAL FLOW TREE

The algorithm "*MatFlow"* is in the centre of the creation of material flow trees ("forward" and "backward") via dynamic pegging.

Input parameters:

- *Object:* Data object for which the material flow tree is built.
- *ExmatArt:* List of material types to be excluded from the tree. (Certain materials, e.g., water, electricity, are not planned and should therefore not be checked for requirements fulfilment.)
- *Direction:* Direction ofthe material flow tree ("forward"= 1, "back"=0).
- *IntramatOnly.* Boolean (0 or 1) indicating whether only intermediate materials are to be included in the tree. Intermediate materials are those materials that are consumed AND produced and thus form the core of the material flow tree. Other materials (like packages) might be consumed, but not produced within the company and are assumed to be in stock.

Output: The return value of the algorithm is a list with the following format: [*Object 1,* [*Path, Fulfilment*], *Object 2,* [*Path, Fulfilment*], ...], where

- *Object X:* Member data object in the material flow tree. One object may appear more than once if it is the dynamic pegging for multiple objects in the material flow tree;
- *Path:* Full path from the root object of the tree (the one for which the tree was called) to the corresponding object;
- *Fulfilment:* Aggregated fulfilment rate (0-1) of all child peggings (successor tree nodes). On the root node it represents the relative amount of the target quantity of the product that will be produced or consumed by all interdependent elements.

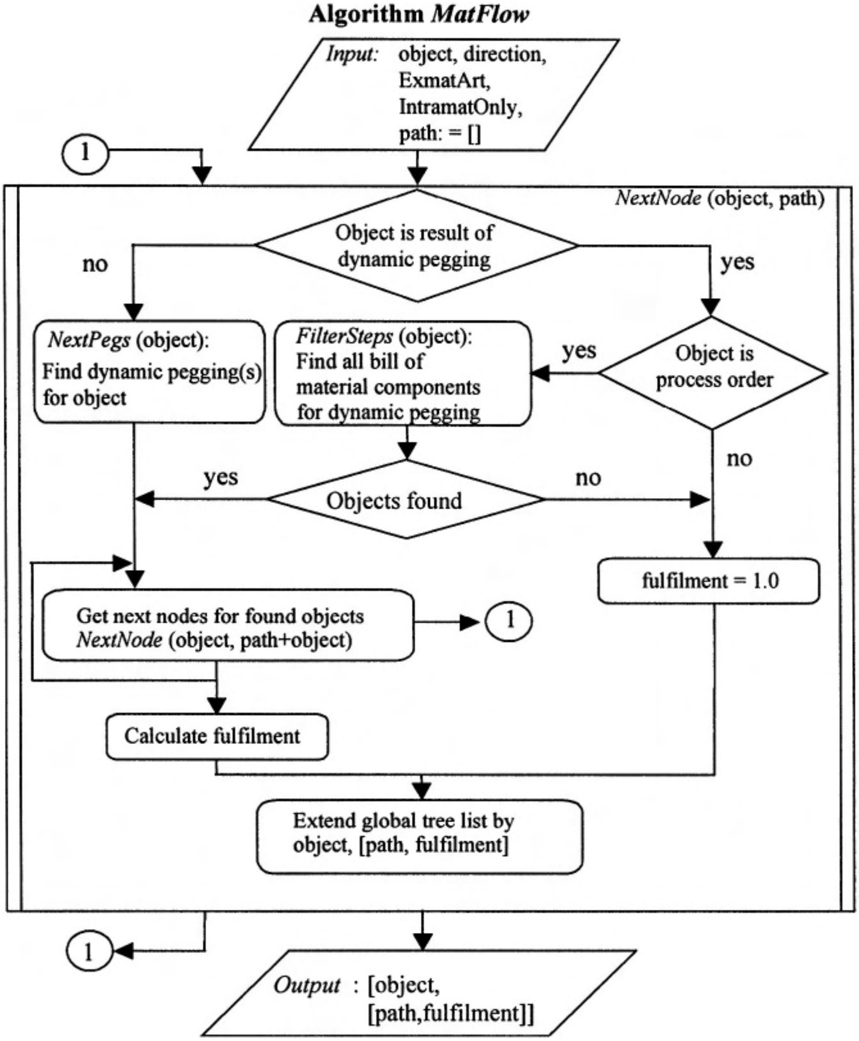

Algorithm *MatFlow*
Input: object, direction, ExmatArt, IntramatOnly, path: = []
1
NextNode (object, path)
Object is result of dynamic pegging
no
yes
NextPegs (object): Find dynamic pegging(s) for object
FilterSteps (object): Find all bill of material components for dynamic pegging
yes
Object is process order
no
yes
Objects found
no
fulfilment = 1.0
Get next nodes for found objects *NextNode* (object, path+object)
1
Calculate fulfilment
Extend global tree list by object, [path, fulfilment]
1
Output : [object, [path,fulfilment]]

Figure 18-6. Algorithm for construction of material flow tree with dynamic pegging

Figure 18-6 provides a generalised flow chart of the recursive material flow tree algorithm. If an object is provided to *NextNode,* the algorithm determines how to find the next nodes. If the provided object was not found by dynamic pegging (e.g., the start object of the material flow tree), the next nodes are located by calling the event list field "DEPENDANTS", which returns the dynamic pegs for the object. For all of these dynamically pegged objects, the function *NextNode* is called again. If the object provided to *NextNode* is a process order that was located by dynamic pegging from a different object, all other entries in the bill of materials are filtered by certain criteria (opposite sign in quantity, *ExmatArt, IntramatOnly*). The BOM components thus located are then passed to another recursion of *NextNode,* in order to find the appropriate dynamic peggings.

At the end of each recursion level the fulfilment is calculated. Since this is done after the recursive function *NextNode* returns, the fulfilment is always computed by taking all of the child branches into account. If the current object was found by dynamic pegging and does not require or produce other materials (not a process order), its fulfilment is simply set to 1.0. If the object is a process order found by dynamic pegging, the fulfilment is the minimum of fulfilments of all child nodes from the bill of materials. If there are no child nodes, fulfilment is 1.0 as well. If the object is not a dynamic pegging, its fulfilment is the sum of the weighed fulfilment of the child branches. After extending a global list with the object and fulfilment, *NextNode* returns.

In order to further illustrate this algorithm, let us consider the small example in Figure 18-7. A customer order 1 requires 1,000 kg of finished product. There are two process orders, each providing 800 kg of finished product, before the customer order has to be delivered. Since dynamic pegging is based on a first-in-first-out approach, process order 1 is completely used to satisfy the customer order, whereas only 200 kg from process order 2 are needed, thus leaving 600 kg available for other use. Therefore 80 % of the customer order is fulfilled by process order 1 and only 2 % by process order 2. Each of the process orders also has requirements in their bill of materials: 800 kg of semi-finished product and 8 pieces of packaging material. Each of these requirements is fulfilled by purchase orders. (In a normal setting the semi-finished product would of course be supplied by another process order.) With regard to the semi-finished product, dynamic pegging for each of the requirements would yield purchase order 1, which fulfils 100 % of each process order, but it is only consumed by 80 %. Regarding the packaging material there is a conflict with process order 2. After process order 1 has already used up 8 of the 10 packages provided by purchase order 2, there are only two left for process order 2. The ten packages of purchase order 3 cannot be used, since the material provision comes in after the requirement. In order to solve the conflict, purchase order 3 would have to be moved to an earlier time.

A material flow tree for the customer has to be built in the "backward" direction (-1), since there is a requirement (negative quantity) for which producers are to be found. When *NextNode* is called with the customer order object, the dynamic peggings will be located by calling the event list field "DEPENDANTS". This will return the two process orders, along with their rate in the fulfilment of the customer order (0.8 and 0.2, respectively). Each of these process orders will be passed to another call of *NextNode,* which will determine all other requirements of these process orders (-800 kg semi-finished product and -8 pcs packaging). For each of these requirements another call of *NextNode* will return the dynamic peggings for purchase order 1 and purchase order 2, respectively.

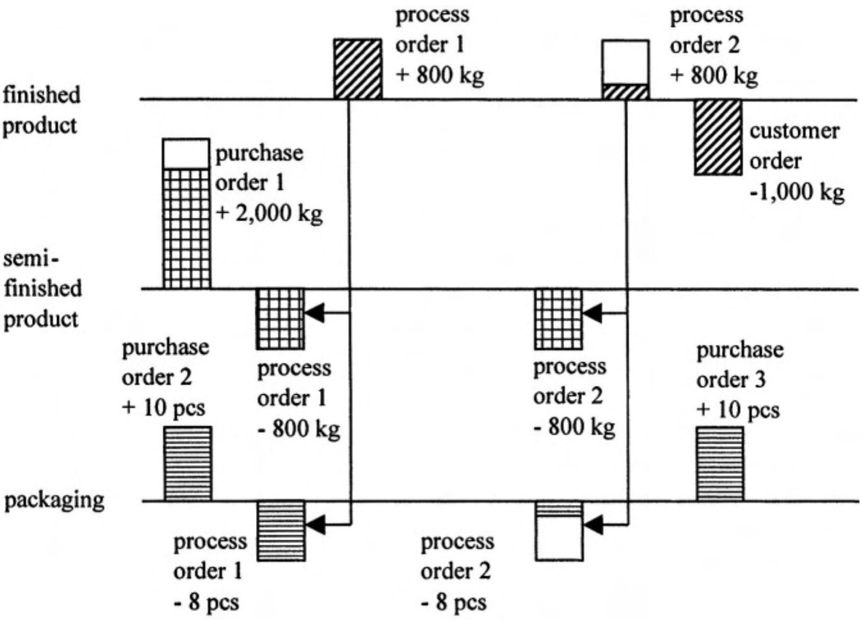


Figure 18-7. Illustrativ example for material flow tree

Due to the recursion fulfilment is calculated “backward” starting from the objects furthest “out” in the tree. Since these are orders with no requirements of their own, *NextNode,* when called for the purchase orders, will assume they are fulfilled at 100 % (1.0). One recursion level up, the process order requirements for the semi-finished product are also fulfilled at 100 % (1.0). For the packaging material only process order 1 will have a fulfilment of 1.0. Dynamic pegging for process order 2 returned purchase order 2, but only with a rate of 0.25 (2 out of 8 pieces) with regard to the full requirement. No other dynamic peggings were found. Therefore the packaging material requirement for process order 2 is only fulfilled at 25 %. On the next recursion level, the fulfilment for each process order is the minimum of the fulfilment of all requirements. Thus process order 1 has a fulfilment of 1.0, process order 2 only of 0.25, so only 200 of the planned 800 kg finished product will be available.

In each recursion step of the algorithm the currently available material pegging is supplemented with the latest calculated information. As a result the whole dynamic material pegging is constructed and the finite recursive algorithm *MatFlow* is stopped. Finally, a browser of the dynamic pegging of material production or material consumption is visualised. The *MatFlow* works actually very fast on real-world data though the volume of master and dynamic data is vast. It took only a few seconds for the material flow construction for the example described in Section 3.3 to be run, although there were 19108 materials, 7016 recipes, 27457 planned orders, 378 process orders over one and a half years, 498 purchase orders and the depth of some material cascades have been equal to 14 (see Figure 18-1). In SCHEDULE++ all this huge information volume can be collected, handled and represented in compact forms for decision-making.

6. FINAL REMARKS

Basic function of SCHEDULE++ is visualisation and representation of data of a selected production (e.g., one of more plants of an described enterprise in one or more R/3 systems). This allows an overview of a whole production situation or supply chain. Based on visualisations experienced decision-makers can create and update typical data objects (e.g., production orders, purchase requisitions and so on) and make decisions in a fast and efficient way. All updates of data objects can be done in a simulation mode and if a satisfying solution is found, all changes can be written back into the R/3 system on users request.

One of the most important advantages of SCHEDULE++ is the simultaneous capacity and material planning. If some production order is moved in the graphic Gantt Chart the impact of the used resources and involved materials will be calculated and shown in the graphic immediately (scheduling of a large numbers of orders with various strategies is supported). Moreover, cascade views of material flows and dynamic peggings of material production and material consumption can be constructed. Planning in a network (simultaneously by several schedulers) as a special case of the limited capacity planning is also possible.

REFERENCES

1. Jänicke W., 2001. Excel Planning Interface for the SAP R/3 System, *Chemical Engineering & Technology,* 24 (9), 903-908.
2. Jänicke W., 2002. Communications - Evolutionary Approach to Production and Requirements Planning in Systems of Chemical Multipurpose Plants, *Chemical Engineering & Technology,* 25 (6), 603-606.
3. Knolmayer G., Mertens P., Zeier A., 2000. Supply Chain Management Based on SAP Systems. Order Processing in Manufacturing Companies, SAP Excellence series, Springer-Verlag, 200 pages.

This page intentionally left blank

Chapter 19

IDENTIFICATION-BASED CONDITION MONITORING OF TECHNICAL SYSTEMS

A Neural Network Approach

Anatoly Pashkevich, Gennady Kulikov,
Peter Fleming, Mikhail Kazheunikau

Abstract: A novel identification-based technique for fault detection and condition monitoring of hydro- and electromechanical servomechanisms is proposed. It is based on neural network analyses of the control charts presenting behavior of the dynamic model parameters. There were derived analytical expressions that allow minimizing impact of the measurement errors on the identification accuracy. The proposed technique has been implemented in a software tool that allows automating the decision-making.

Key words: condition monitoring, identification, control charts, neural networks.

1. INTRODUCTION

Increasing complexity of technical systems requires detecting and isolating failures at an early stage, to avoid costly financial losses from unplanned unit shut down and incorrect fault detection. In some application cases (aircraft gas-turbine engines, for instance), it is even a vital issue [9]. For this reason, the area of condition monitoring and fault detection has received considerable attention from industry and research community [2,3,13]. However, as it was stressed in [5], no single method is capable of meeting all the requirements of a good diagnostic system and a hybrid framework involving classical control charts and intelligent reasoning tools is recognized as an attractive alternative.

At present, the most developed and frequently used monitoring techniques are based on the Shewart charts, which are aimed at detecting of an abnormal change in the underlying distribution (of the quality variables or other key variables). The basic assumption behind them is that a process, which is subject to only *natural variability,* will remain in a state of statistical control unless a special event occurs [11]. However, in the case of technical system monitoring, the *correlation* between the original variables should be considered, to minimize the chance to miss a system malfunction [6]. It leads to *identification-based* methodology, which should be also robust in respect to the measurement noise.

The control chart technique represents several statistical hypothesis testing procedures aimed at detecting of a special event. In practice, it is often based on the visual judgment of an operator, who recognizes unusual patterns. To assist him/her, a number of supplementary rules, like zone tests or run rules were developed [7]. The run rules are based on an assumption that a pattern has a very low probability for a completely random scattering of points around a mean. One of the main problems with using the run rules is that simultaneous application of them may increase the probability of wrong fault detection (i.e., the Type I error), and a number of authors consider the run rules as not a very efficient tool for such application. For this reason, several other techniques were deployed, including expert systems, neural networks (NN) and heuristic algorithms. Although some results were promising, a common problem reported in several studies was false alarm generation [1,4,8,10,12]. This encourages for further research taking into account particularities of specific applications.

In this chapter, it is proposed a neural network based technique, which employs a pattern discrimination algorithm to recognize unnatural control chart patterns for industrial robotic manipulators and aircraft gas-turbine engines equipped with incremental encoders. In contrast to the conventional approach, which rely on analysis of direct measurements taken at regular intervals and comparing against the reference level, the proposed technique deals with dynamic model parameters analysis, which are estimated via the on-line identification. There is also proposed a novel identification algorithm, which is based on incremental encoder reading and digital filtering of the measurement noise. To tune this algorithm, there are derived analytical expressions and numerical routines that allow computing the algorithm parameters and minimize the impact of the measurement errors. The proposed pattern discrimination algorithm is based on several neural networks trained for this specific recognition task. Numerical results show that the false recognition problem has been effectively addressed. The developed technique has been carefully validated via a computer simulation and also via some real-life tests. To implement the proposed technique, a software tool has been developed that allows processing a set of files with “raw” data, identification of the model parameters, and visualizing/analyzing the identification results in respect to detection of failures.

The remainder of the chapter is organized as follows. In Section 2, the identification technique is being developed and optimized concerning the measurement noise and round-off errors. Sections 3 and 4 are devoted to the identification data analysis and contain description of the relevant software tools and the decision rules. Section 5 focuses on industrial case study and future applications, while conclusion summarizes the presented results.

2. IDENTIFICATION OF MODEL PARAMETERS

Typical hydro/electromechanical transmission, which is widely used in both aircraft and industrial equipment, may be described by the following equation

$$\dot{p}(t)=v(t);\quad T\cdot\dot{v}(t)+v(t)+m\cdot\operatorname{sgn}(v(t))=k\cdot u(t),\tag{1}$$

where $p(t)$ is the output variable (shaft angle); $v(t)$ is its time derivative (shaft speed); $u(t)$ is the input control variable (control code); k is the transmission gain; T is the model parameter (electromechanical time constant); m is the disturbance amplitude. It should be stressed that, in this study, values of T and m are the main indicators of the object state and source data the for the condition monitoring.

Taking into account the time sampling, the original model can be converted into the discrete-time form

$$v_{k+1}=\mu\cdot v_k+(1-\mu)\cdot(k\,u_k-m);\quad v_k=(p_{k+1}-2p_k+p_{k-1})/2T_0,\tag{2}$$

where $\mu=\exp(-T_o/T)$; T_0 is the sampling time. To identify the model parameters, first let us assume that the transmission gain k is known and values of the shaft speed v_k can be extracted from the experimental data with the sufficient accuracy. In this case, the basic identification equation may be written as

$$A_k+m=(B_k+m)\cdot\mu;\quad A_k=v_{k+1}-k\,u_k;\quad B_k=v_k-k\,u_k.\tag{3}$$

To obtain expressions for estimation of the unknown model parameters (μ,m), let us define the quadratic objective function

$$F(\mu,m)=\sum_{k=1}^{n}[(A_k+m)-(B_k+m)\cdot\mu]^2,\tag{4}$$

and set its derivatives equal to zero. Solution of the corresponding system of equation yields

$$\mu = \frac{n\sum_{k=1}^{n} A_k B_k - \sum_{k=1}^{n} A_k \sum_{k=1}^{n} B_k}{n\sum_{k=1}^{n} (B_k)^2 - \left(\sum_{k=1}^{n} B_k\right)^2}; \qquad m = \frac{\mu \sum_{k=1}^{n} B_k - \sum_{k=1}^{n} A_k}{n(1-\mu)}. \tag{5}$$

where n is the number of input/output samples.

If the transmission gain k is treated as an unknown, the dynamic model (2) can be considered as the conventional linear regression $v_{k+1} = a \cdot v_k + b \cdot u_k + c$ where $a = \mu$; $b = (1-\mu)k$; $c = -(1-\mu)m$ and the corresponding identification problem can be solved via the pseudoinverse of Moore-Penrose:

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} \sum_k v_k^2 & \sum_k v_k u_k & \sum_k v_k \\ \sum_k v_k u_k & \sum_k u_k^2 & \sum_k u_k \\ \sum_k v_k & \sum_k u_k & N \end{pmatrix}^{-1} \cdot \begin{pmatrix} \sum_k v_{k+1} v_k \\ \sum_k v_{k+1} u_k \\ \sum_k v_{k+1} \end{pmatrix} \tag{6}$$

Then the model parameters may be extracted from *a, b, c* using expressions

$$\mu = a; \quad m = -c/(1-a); \quad k = b/(1-a). \tag{7}$$

However, as follows from the separate research, the latter expressions are less robust in respect to the measurement noise compared to the expression (5). For this reason, it is reasonable to decompose the identification procedure, by separating estimation of the transmission gain (in the steady-state mode) and identification of the dynamic parameters *T* and *m,* which are used for condition monitoring and fault diagnosis.

To apply expressions (5), (7) for the identification, it is required to have arrays of the transmission velocities and the corresponding control inputs $\{v_k, u_k \mid k = 1:n\}$. However, most of transmissions are equipped with incremental encoders, which allow obtaining the position code and control $\{p_k, u_k \mid k = 1:n\}$ only. For this reason, the velocity must be estimated using certain computing schemes. It is obvious that the simplest velocity estimate $\hat{v}_i^{(1)} = (p_{i+1} - p_i)/T_0$ is inaccurate and very sensitive to the sampling noise. To show it, let us assume that the measurement errors are modelled as the white

Gaussian noise with zero mean and variance σ^2, then the standard deviation of the velocity estimate is $\delta\left[\hat{v}^{(1)}\right]=\sigma\sqrt{2}/T_0$. Under the assumptions made, the value of σ depends on two main factors. The first of them (level sampling) may be presented as the white noise with the s.t.d. $\sigma_1=\Delta/\sqrt{12}$, where Δ is the quantization step. The second factor (time sampling) causes delays in position sensing, which may be described as the white noise with the parameter $\sigma_2=v_{\max}\cdot T_0/\sqrt{12}$, where v_{max} is the maximum velocity. Assuming statistical independents of these factors, the variance of the measurement noise may be expressed as $\sigma^2=\sigma_1^2+\sigma_2^2$. As follows from the related research, the resulting error of the velocity estimation may be rather high and essentially influence on the identification results for the model (1). For instance, for a typical transducer with the parameter $\Delta=2\pi/1024$ and the maximum velocity $v_{\max}=10\, inc/tic$, the corresponding noise parameter is $\sigma\approx 2.9\, inc$ and the relative velocity estimation error is $\delta\left[\hat{v}^{(1)}\right]/v_{\max}\approx 40\%$ (see Figure 19-1). Therefore, the identification of the dynamic model parameters must relay on the more accurate velocity data.

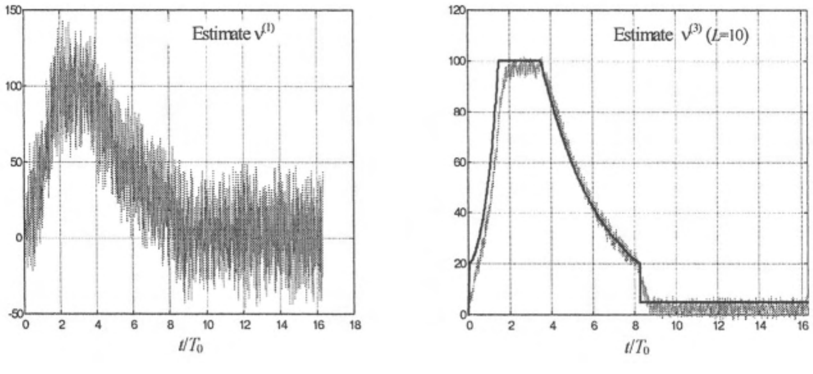


Figure 19-1. Comparison of the velocity estimation techniques

To increase the accuracy of the velocity estimation, let us apply an estimation technique for the time window $\left[t-LT_0;\;\; t+LT_0\right]$. The simplest estimate of this type $\hat{v}_i^{(2)}=\left(p_{i+L}-p_{i-L}\right)/2LT_0$ relies on the boundary values of $p(t)$ and yields the error $\delta\left[\hat{v}^{(2)}\right]=\sigma\sqrt{2}/2LT_0$. Further error reduction can be achieved by using a linear spline for smoothing of $p(t)$, i.e.

$$v_i^{(3)}=\arg\min_{v,a}\sum_i\left[a+vt_i-p_i\right]^2 . \qquad (8)$$

Differentiating this expression with respect to *a, v* and solving corresponding equations yields

$$\hat{v}^{(3)} = \bar{S}_{pt}/\bar{S}_{tt}\,, \tag{9}$$

where

$$\bar{S}_{pt} = \sum_k \left(p_k - n^{-1}\sum_k p_k\right)\cdot\left(t_k - n^{-1}\sum_k t_k\right);$$

$$\bar{S}_{tt} = \sum_k \left(t_k - n^{-1}\sum_k t_k\right)^2.$$

To evaluate these estimates, let us assume that $p(t) = vt + \xi$, where ξ is the white noise with the parameters $(0, \sigma)$ and the time window is $t \in [-LT_0; LT_0]$. For this window, the sums are equal to

$$S_{pt} = vT_0^2 \cdot \sum_{i=-L}^{L} i^2 + T_0 \cdot \sum_{i=-L}^{L} i\xi_i\,, \quad S_{tt} = T_0^2 \cdot \sum_{i=-L}^{L} i^2\,, \tag{10}$$

so the velocity estimate may be presented as follows

$$\hat{v}^{(3)} = v + 3\sum_{i=-L}^{L} i\xi_i \Big/ T_0\left(2L^3 + 3L^2 + L\right). \tag{11}$$

Hence, it is the unbiased estimate and the standard deviation of $\hat{v}^{(3)}$ is computed as

$$\delta\left[\hat{v}^{(3)}\right] = \sigma / vT_0\sqrt{2L^3 + 3L^2 + L}\,. \tag{12}$$

As follows from the relevant analysis (see Figure 19-2), reasonable for our application accuracy $\delta\left[\hat{v}^{(3)}\right] \leq 0.004$ is achieved for the time window parameter $L \approx 20$. It should be also stressed that in contrast to the conventional identification techniques, which deal with the linear approximation of time series, the proposed method relies on highly non-linear relation between the model input and output.

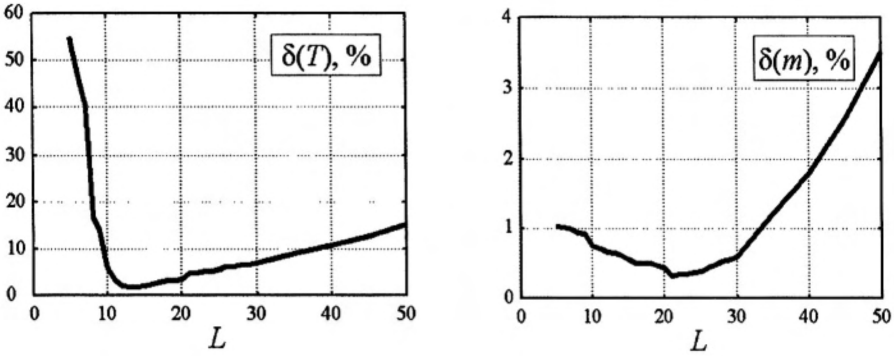


Figure 19-2. Estimation errors δ(*T*) and δ(*m*) for different values of *L*

3. ANALYSIS OF IDENTIFICATION DATA

Following to the basic concepts of the applied statistics, let us assume that the variation of the identified model parameters can be partitioned into two components. The first one (natural process variation) is caused by naturally occurring fluctuations or variations inherent in the object, while the second component (special cause variation) is caused by some extraordinary occurrences. Therefore, to detect unusual object behaviours, the control chart technique can be applied, which is widely used in the statistical process control [5]. However, a specific nature of the input data should be also taken into account, as ones are mixed with additional noise caused by the identification and velocity estimation procedures.

After comparison of different types of the control charts, there were selected several types that are suitable for the condition monitoring. As follows from the industrial case studies, the most reliable fault detection ensure the X, R, S – charts and their combination, while the mowing window technique yielded a number of false fault warnings. To analyse the identification data, the **3σ-range** of the identified parameters was divided into there equal zones (A,B,C) and ten decision rules were selected, violating which posses the probability of much less then 0.01 for a usual object state. These rules are as follows: “Out of Control Limits”, “Run above/below the centerline” (7 of 7, 10 of 11, 12 of 14, and 14 of 16 points), “Run in zone C” (15 points of 15), “Avoidance of zone C” (4 points of 5), “Points in zone A or outside” (2 points of 3), “Linear Trend” (7 successive points), “Oscillatory Trend”. It should be stressed, that all these rules must be applied simultaneously to decrease the probability of false alarms.

The relevant software tool, Model Parameter Analyst, is a 32-bit Windows NT/2000 application which incorporates the latest software development technologies and techniques. Conventional Windows editing features enhance its efficiency and make it easy to view, sort and rearrange data measurements. It future, it will be enhanced with the client-server features to get access to the relational databases and enlarge quantities of data to be stored and analysed. Large sized data plots keep each point information, providing required details of the point in question. Auto-linking ties data views to hierarchy windows, so when a user scrolls through data plots, alarm details change automatically to show the highlighted point. The software tool provides intelligent advisories to operations and maintenance personnel by analysing the dynamic model parameters evolution.

4. RECOGNITION OF ABNORMAL PATTERNS

4.1 Neural Network Structure

The feedforward network is used for recognition of abnormal patterns. Multiple layers of neurons with nonlinear transfer function allow the network to learn nonlinear relationships between the input and output vectors. The proposed modification of the network is represented by four layer structure (Figure 19-3), which input layer contains 21 neurons used as input data from control chart (20 consecutive points and the mean *M).* The last layer consists of a single neuron, which output is scaled within [0,1], where 1 and 0 represent the data corresponding to the normal or abnormal patterns. There are two hidden layers, which contain 60 and 40 neurons respectively. For the hidden and output layers activation, it is used the hyperbolic tangent function.

4.2 Generation of Training Sets

The Monte-Carlo simulation approach was used to generate required sets of control chart patterns for both training and testing examples. Several basic patterns typically found in control chart were considered, namely: the upward shift, downward shift, upward trend, downward trend, cycle, systematic and mixed trend-cycle patterns, as well as normal pattern of random variation. The abnormal patterns were generated using the expression: $x_k = n_k + d_k$, where x_k is the sample value at the time instant *k;* n_k is the common cause variation following a normal distribution with the mean M and the standard deviation σ; and d_k is a disturbance.

In the case of the shift pattern, the disturbance is defined by the equation $d_k = us \cdot s_n$, where u=0 before the shift and u=1 after the shift; s_n=1 for the upward shift patterns and s_n=-1 for the downward shift ones; s is the shift magnitude $(1 \le s \le 3)$. For the trend, the disturbance equation is $d_k = ds_n \cdot k$, where d is the slope; s_n=1 for the upward trend patterns and s_n=-1 for the downward ones. For the cycle pattern, the equation is $d_k = a \cdot \sin(2\pi k / \Omega)$, where a is the cycle magnitude $(1 \le a \le 3)$, Ω the cycle period. For the systematic pattern, the corresponding equation is $d_k = m \cdot (-1)^k$, where m is the pattern magnitude $(1 \le m \le 3)$, defining the fluctuation above or below the process mean. For the mixed trend-cycle patterns, the equation is $d_k = d \cdot k + a \cdot \sin(2\pi k / \Omega)$.

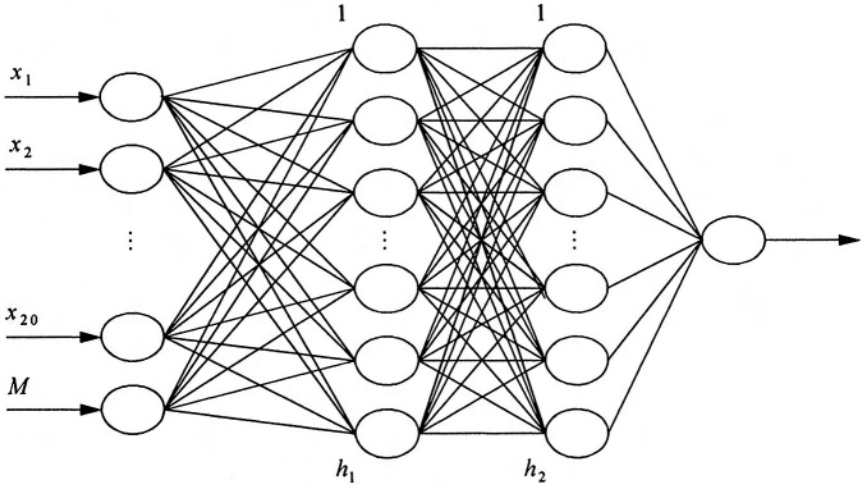


Figure 19-3. Neural network for control chart analysis

The backpropagation algorithm has been used for the neural network training. During training, the weights of the network were iteratively adjusted to minimize the mean square error (MSE), i.e. the average squared error between the network output and the target output. The algorithm is based on the gradient of the performance function to determine how to adjust the weights to minimize the performance measure. The gradient is computed applying the backpropagation technique, which involves computation backward through the network. The batch mode of the backpropagation is used, i.e. all of the inputs are applied to the network before the weights are updated. For the network training, 3800 examples were generated, among them 1000 normal, 200 upward shift, 200 downward shift, 200 upward trend, 200 downward trend, 200 cycle, 200 systematic and 200 mixed trend-cycle, with different value of the mean *M*. The learning rate was set to 0.5. The connection weights were updated until the convergence condition $\varepsilon \le \varepsilon_{\max}$ or $i \le i_{\max}$ satisfied, where ε is the MSE value, and *i* is the iteration number.

Table 19-1. The network training result

h_1	h_2	ε	Execute time, s.	P,%
20	20	0.00162	300	89.5
25	25	0.000132	330	87.5
30	30	0.00035	408	91.6
35	40	0.00175	420	90.0
40	40	0.00052	480	91.1
45	40	0.00015	510	93.6
50	40	0.00015	528	93.7
60	40	0.00018	600	94.0

4.3 Simulation results

During simulation, several feedforward networks with different dimension of the hidden layers h_1 and h_2 have been trained on control charts patterns. The network training was implemented using MatLab 6.0 package. The backpropagation algorithm parameters were: $\varepsilon_{\max}=10^{-5}$ and $i_{\max}=3000$ (see Figure 19-4 for a training session example). For the networks testing, 140 abnormal and 50 normal control chart patterns with different M values were generated and applied to the networks input. The training and testing results are presented in Table 19-1, which shows the reached MSE value and overall percentage of the correctly recognized control chart patterns from the testing set P. As follows from them, the minimal value of the MSE $\varepsilon=1.5\cdot10^{-4}$ was reached in the case of h_1=50 and h_2=40 neurons in hidden layers. But as it can be seen from the table, in the case of h_1=60 and h_2=40 neurons, 94% test examples were correctly recognized, and an average percentage of correctly recognized patterns is about 90% for all case studies.

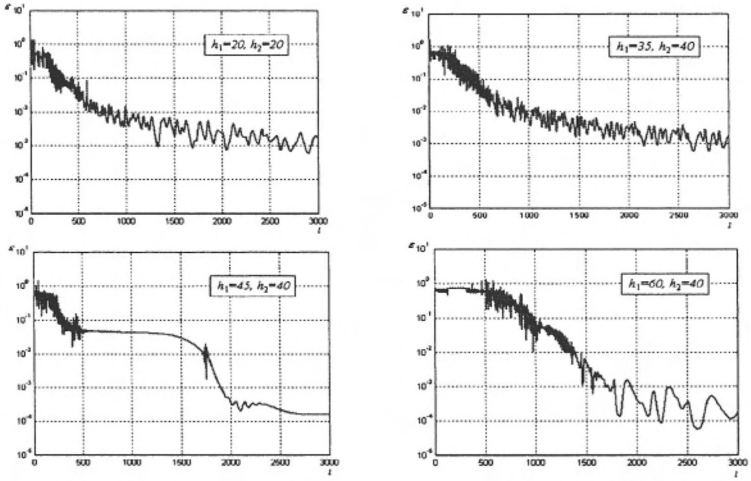


Figure 19-4. Neural network training sessions for different values of h_1 and h_2

5. INDUSTRIAL APPLICATIONS

The developed technique and software tool have been successfully applied for the on-line failure detection and predictive maintenance of an industrial robotic system, which is incorporated in a manufacturing line for fabrication of printed circuits. The system consists of 5 gantry robots that serves over 200 workstations with the motion speed over 1.2 m/s. The robots are equipped with incremental encoders that are used to compute speed and position feedback for the local controller. The encoder readings, together with the controller output code, are used for the identification of the dynamic model parameters that are treated as indicators of failures. The identification is carried out in accordance with a user-specified time window and the output is stored in a database, which also contains log-files and control actions history. For each robot, the identification data are analyzed using the developed decision rules and, after violation any of them, it is generated a warning, and the identification data are presented to an operator to make a final decision on current condition of the robot. Also, such visualization allows making a well-grounded decision on predictive maintenance intervals and is useful for tuning of the controller parameters. The obtained results were approved by manufacturing engineers who at the moment are adopting such an innovative approach to the maintenance as ODR (Operator Driven Reliability).

Another application will deal with the condition monitoring and fault diagnosis of gas turbine engines (GTE), which is a vital issue in the flight safety. Gas turbines are also critical to the operation in most industrial plants, and their associated maintenance costs can be extremely high. To reduce those costs, the power industry has moved sharply toward condition-based maintenance and monitoring. In this approach, intelligent computerized systems monitor gas turbines to establish maintenance needs based on the turbine's condition rather than on a fixed number of operating hours. Such approach significantly cuts costs and improves performance by using control-system information to perform gas-turbine condition monitoring.

This application is being developed in the frames of INTAS project 2000-757, which combines experience of seven international teams from the UK, Russia, Ukraine, Belarus and Portugal. The project is aimed at creating new methods for condition monitoring by integrating several AI technologies and techniques from other application areas, such as adaptive control and robotics. The project output is a prototype electronic unit for condition monitoring and fault diagnosis of GTE and their control systems, which will be verified both on test-bed facilities and aircraft flights in Ufa (Russia).

6. SUMMARY

Early diagnosis of process faults can essentially reduce amount of productivity loss during an abnormal event. Moreover, in some application cases (aircraft gas-turbine engines, for instance), it is a vital issue. In this chapter, it is presented a novel identification algorithm for hydro- and electromechanical servomechanisms used in such equipment, which is based on incremental encoder reading and digital filtering of the measurement noise. There were derived analytical expressions and numerical techniques that allow computing the algorithm parameters, which minimize the impact of the measurement errors and round-off of the identification routines. To analyse the identification data, it has been applied the neural network, which allows distinguishing step change of the model parameter in contrast to usual variation of the identification procedure caused by the random factors. The developed technique has been implemented in the software tool and carefully validated via computer simulation and real-life tests in robotic manufacturing system.

ACKNOWLEDGEMENTS

The authors appreciate the financial support of the INTAS through the project 2000-757.

REFERENCES

1. Anagun A. S., 1998. A neural network applied to pattern recognition in statistical process control. *Proceedings of the 23rd Int. Conf. on Computers and Ind. Engineering,* Chicago, IL, 185–188.
2. Basseville M. and Nikiforov I.V., 1993. *Detection of abrupt changes: theory and applications,* Englewood Cliffs, NJ: Prentice-Hall.
3. Chang S. I., 2000. An Integrated Scheme for Process Monitoring and Diagnosis. *Proceedings of ASQC 49th Annual Quality Congress,* Cincinnati, OH, 725-732.
4. Cheng C.S., 1997. A neural network approach for the analysis of control chart patterns. *International Journal of Production Research,* 35, 667–697.
5. Dash S. and Venkatasubramanian V., 2000. Challenges in the industrial applications of fault diagnostic systems. *Proceedings of the conference on Process Systems Engineering,* Keystone, Colorado, 785-791.
6. Frank P., 1990. Fault diagnosis in dynamic systems using analytical knowledge-based redundancy- a survey and some new results. *Automatica,* 26(3), 459-474.
7. Grant E.L. and Leavenworth R.S., 1996. *Statistical quality control,* New York:McGraw-Hill.
8. Hwarng H.B. and Hubele N.F., 1993. X-bar control chart pattern identification through efficient off-line neural network training. *IIE Transactions,* 25, 27–40.

9. Kulikov G.G., Breikin, T.V., Arkov V.Y. and Fleming, P.J., 1999. Real-time simulation of aviation engines for FADEC test-beds. *Proceedings of the International Gas Turbine Congress,* Kobe, Japan, 949-952.
10. Lucy-Bouler T.L., 1993. Application to forecasting of neural network recognition of shifts and trends in quality control data. *Proceedings of WCNN'93—World Congress on Neural Networks,* Portland, UK, vol. 1, 631–633.
11. Montgomery D.C., 1996. *Introduction to Statistical Quality Control,* John Wiley and Sons, Inc., New York.
12. Pham D.T. and Oztemel E., 1994. Control chart pattern recognition using learning vector quantization networks. *International Journal of Production Research,* 32, 721–729.
13. Tsung F., 2000. Statistical Monitoring and Diagnosis of Automatic Controlled Processes Using Dynamic PCA. *International Journal of Production Research,* 38, 625–637.

This page intentionally left blank

Chapter 20

SIMULATION OF DISTRIBUTED INDUSTRIAL SYSTEMS

A multi-agent tool

Stéphane Galland, Frédéric Grimaud,
Philippe Beaune, Jean-Pierre Campagne

Abstract: We are located in the context of the industrial system simulation, which are complex and distributed in operational, informational and decisional terms. In this chapter, we present the problems and a methodological solution. This methodology is based on the systemic approach and on multi-agent systems. It allows the modelling of distributed industrial systems such as enterprise consortiums. Moreover, it proposes a software platform architecture whish is currently instanced with Arena and dedicated agents.

Keywords: Industrial System, Discrete-Event and Distributed Simulation, Multi-Agent Systems, Decision-making process.

1. CONTEXT AND PROBLEMS

The simulation is a tool adapted to the studying of modern industrial problems and more precisely the dynamic behaviour of industrial systems [4]. In this context, the support of the new industrial organizations is particularly focused. An example of new industrial organization is the enterprise consortiums, which is a whole of companies related the ones to the others by a cycle of production. The bond is neither legal, nor structural; it has often the form of simple agreements. These companies have in common a powerful system of functional cooperation [2].

Even if the simulation is powerful, some problems always exist. This chapter is concentrated around four of them. First, the simulation tools are still seldom packaged with a dedicated methodology. They have a strong influence to the designers' point of view. For example, Arena® and Simple++® offer two modelling views which are similar and different in the same time: the modelling concepts are similar but they are not used or defined in exactly the same way. This formalisation problem is partly solved by existing methodologies

The second problem is the poor support of the component-based or modular modelling.

Next, the strong relationship between the physical, the informational and the decisional aspects of an industrial system is also highlighted. Currently, the simulation models include these two kinds of flows. But they don't highlight each of them. Then the understanding is still difficult according to the necessity to mentally distinguish them. Another example is when a designer wants to update the decisional (e.g. the management policy). Then, in most of cases, he must remodel and rewrite all the models to include this change.

Finally, the last problem is about the difficult to model the new industrial organizations, such as enterprise consortiums or virtual enterprises. This problem has two sides: the modelling and the simulation. The modelling of distributed industrial systems is not naturally supported by the tools. Moreover, all the tools do not accept to simulate on a computer network. This constraint is for instance introduced by the confidentiality imposed by the consortium members.

To solve these different problems, a methodological approach is proposed: MaMA-S (Multi-Agent Methodological Approach for the Simulation of industrial systems) [6,7]. It offers a modelling framework that is independent of any software platform (simulation tool or multi-agent system). In the rest of this chapter, the major concepts attached to this methodology are presented. More precisely, the life cycle and the major propositions on this methodology are explained.

2. METHODOLOGICAL APPROACH

This section presented the major propositions about MaMA-S. See [6,7] for more details.

2.1 Life cycle of the MaMA-S models

The development of a methodological approach passes by a first major stage: the definition of the life cycle of the models. This section is devoted to the definition of this life cycle for the models of distributed industrial systems. Starting from the assets of the software engineering and the works already completed in the field of simulation, an extension of the existing approaches is proposed to take the new enterprise organizations into account.

Figure 20-1 illustrates the life cycle used by MaMA-S. Their contributions are restricted to the adaptation of the *specification,* the *design,* and the *implementation.* Two special stages are also included: the *methodological guidelines* and the *coherence checking.*

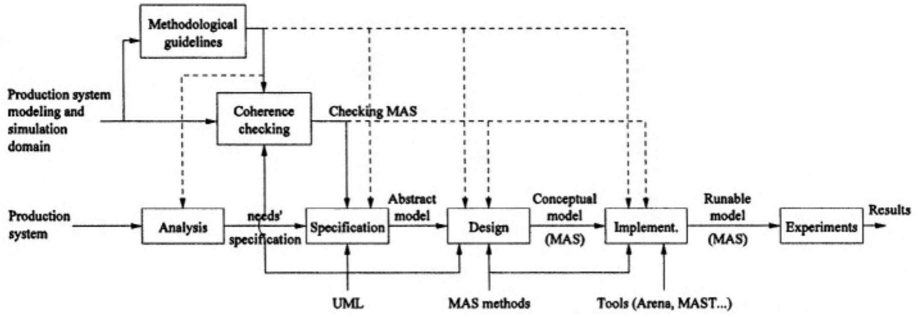


Figure 20-1. Life cycle of the MaMA-S models

The methodological guidelines are written during the stage of the same name. It permits to specify the principles of the methodology (life cycle, modelling elements, methods...). The guidelines are, at the same time, the specifications of and a user guide. Currently, they are limited to the specification of from [6,7]. It will evolve according to the progresses of the works on MaMA-S.

The coherence checking aims to check the coherence of the different simulation models. This stage is not presented in this chapter. You could read [6,7] for more details.

The other adapted stages are presented in the following sections.

2.2 Phase of Specification

The phase of specification is crucial in MaMA-S. Indeed, it corresponds to the moment when the first formally expressed model must be produced. In this section, the methodological bases and the subjacent principles of the abstract model's specification are presented.

The modelling elements are used within the framework of the specification for the creation of an abstract simulation model. This building must be carried out starting from the information collected and exposed in the needs' specification. This methodological approach considers that the distributed production system can be broken up according to the systemic approach proposed by Jean-Louis Le Moigne [8]: an operational subsystem, an informational subsystem and a decisional subsystem. Some modelling concepts are proposed for each of these subsystems.

a) Physical subsystem

The physical subsystem is the whole of the industrial infrastructures of the modelled system. The basic concepts supported by are partly from [1]:

- **Composition:** the concepts of model and sub-model;
- **Critical resources:** the resources which can stop the physical flow.
  - active resources: used to realize an activity (processing units, human resources, transportation means...)
  - passive resources: used by the active resources to realize these activities.
- **Queue:** it represents an ordered list of physical entities waiting for a specific event;
- **Structural modelling of the physical flow:** a set of additional modelling elements that permit to define the paths used by the physical entities (links, junctions, forks, jumps, exit points and entry points).
- **Distribution:** the whole of modelling elements that allows the definition of a distributed model. They are defines in [6,7].

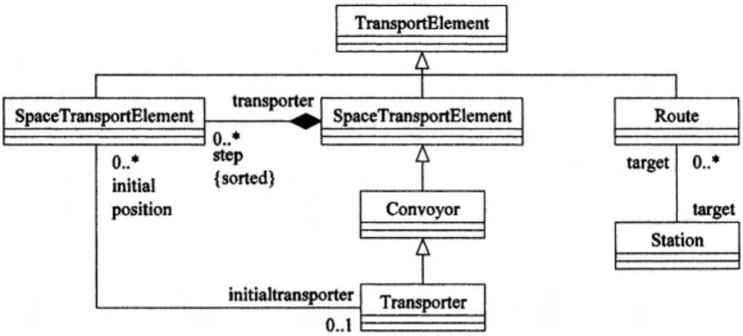


Figure 20-2. Part of the physical sub-system meta-model

The modelling artefacts are defined in an extension of the UML meta-model. Figure 20-2 illustrates a part of the physical subsystem meta-model. It corresponds to the definition of the modelling elements for the transport means. The roads (Route) permit to reach a destination. The elements of type TransportElement contain a stochastic law for the transport duration. Thus, a road supports the temporal aspect of the transport. The two other kinds of transport means (conveyors and transporters) extend the concept of road by including a spatial aspect. The difference between a conveyor and a transporter is the limitation of the transportation resources in the second.

To permit an easier design of the simulation model, a graphical language attached to each object defined in the meta-model is proposed in [6,7].

To illustrate the operational subsystem modelling, we propose to create a model of a simple distributed system: a consortium of three enterprises E_1,

E_2 and E_3. The objective of this consortium is to produce movement-detecting cameras.

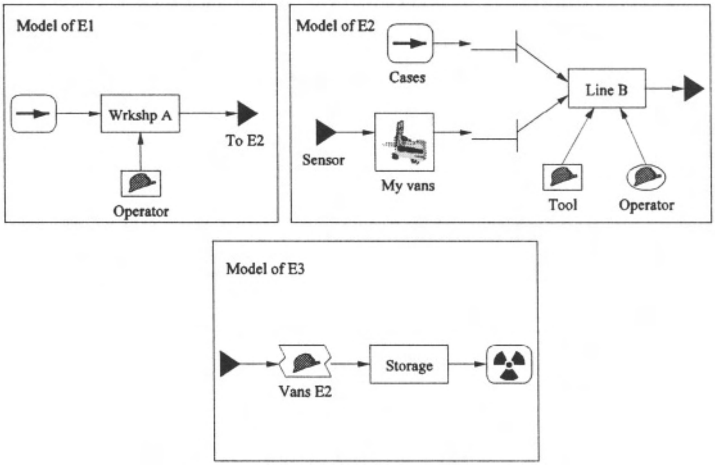


Figure 20-3. Example of a physical sub-system model

The enterprise E_1 produces sensors in its workshop *A*. They are sent to the second enterprise. This last assembles in its workshop *B* the sensors with the cases which are locally manufactured. Then, E_2 forwards the resulting detectors to the third consortium member which must only store the final products. The transport between these three enterprises is exclusively carried out by the transport services of E_2. Additionally, the workshop *A* uses a critical resource: an operator, and the workshop *B* uses two resources: a critical resource which permits to assemble the cameras, and a passive resource representing an supervisor. Moreover, the agreements between the consortium members specify that E_1 does not know how the sensors are transported, and that E_2 does not force E_3 to use its transport services. These considerations enable us to put the transportation modelling artefacts in the models of E_2 and E_3. Figure 20-3 graphically illustrates the three resulting models.

b)Decisional subsystem

The decisional subsystem is the whole of the organisational structures and decision-making processes of the industrial system. The UML meta-model of MaMA-S defines a language that permits to describe the relational structures between the *decision-making centres.* The centres can take operational, tactical or strategic decisions. The relationship between the centres can be hierarchic or cooperative. This point of view is issued from the works on the organisational structures in industrial systems and in multi-agent systems.

Each decision-making centre includes at least one *behavioural model.* Each of them could be a protocol based on state-transitions, on stimuli or on both of them.

Let us take again the example of the consortium of three enterprises. Here, we are interested exclusively in the decisional subsystem modelling. First, the focus is on the management policy. The enterprise E_3 is in direct relation with the "market". Each time its stock of cameras does not permit to answer to an order, this enterprise sending a production order to E_2. This one has pulled flow management policy. For each production order coming from E_3, it creates a corresponding production order to E_1. This last launches the production of the number of sensors claimed by E_2. The production process uses a pushed flow management policy. Figure 20-4 illustrates the decisional models for each of the three members of the consortium.

Each decisional centre has its own behaviour (the used language is defined in [6,7]). For instance, the production controls of E_1 and E_2:

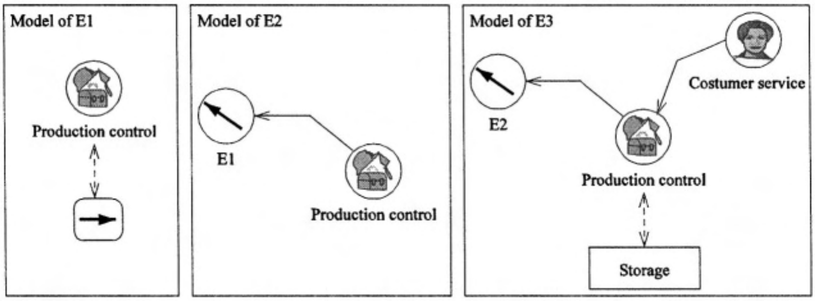


Figure 20-4. Example of a decisional sub-system model

- **Production control of E_1:**
  This centre generates a physical entity for each sensor having to be produced. Its behaviour can be defined as follow:

```
CONTEXT "Production control"
WHEN RECEIVE "Production order"
WITH PARAMS ( "size of the PO" )
THEN
   i = 1;
   WHILE ( i <= "size of PO" )
   DO
      OPERATION( generate-entity,
         INFORMATION(entity) ) ;
      i = i + 1 ;
   DONE
END
```

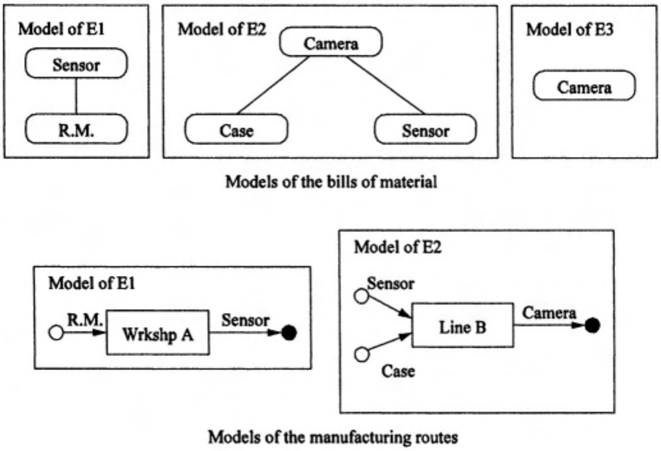
– **Production control of E_2:**
This centre generates a production order of sensors towards E_1 for each order coming from E_3:

```
CONTEXT "Production control"
WHEN RECEIVE "Production order"
WITH PARAMS ( "size of the PO" )
THEN
   SEND TO E1
      TYPE ORDER
      NAMED "Production order"
      DATA "size of PO" *
         INFORMATION(
            bill-of-material-camera
            "quantity of sensors" ) ;
END
```

c) Informational subsystem

The informational subsystem contains the information used by the two other subsystems. In the MaMA-S modelling language, the concepts of *bill of material, manufacturing routing* and *entities* (physical or decisional) are defined.

Let us take again the example of the already presented consortium. Consider the models of bills of materials (Figure 20-5). Each enterprise has its own vision of the products. However, the sensor used by E_2 is in fact the definition being in the model of E_1. This is a simple usage example of a distant product definition. The bill-of-material model of E_3 is the same as the model of the enterprise E_2.



Models of the bills of material

Models of the manufacturing routes

Figure 20-5. Example of an informational sub-system model

Consider now the manufacturing routing. In our example, only the first two enterprises must define a manufacturing routing model. Indeed, E_3 does not carry out any transformation on the products. Figure 20-5 illustrates the graphical representation which is proposed within MaMA-S. Thus, in E_1, the raw material (`R.M.`) is transformed into sensors by the workshop *A,* and in E_2, the cameras are obtained starting from the assembly of the sensors and the cases. Note that, for each manufacturing routing model, the treatment units must be associated to the corresponding processing units from the operational subsystem (they must have the same name). In addition, a treatment must define a whole oftimes necessary to model the processing durations.

2.3 Phase of Design

The conceptual model is a multi-agent system (MAS) model [5] that corresponds to a translation of the abstract model shown in the previous section. It describes the structural organization ofthe agents. But, it does not force to use a particular multi-agent platform or a particular simulation tool. The only one constraint is that it must respect a specification according to the approach "Vowels" (or AEIO) [3].

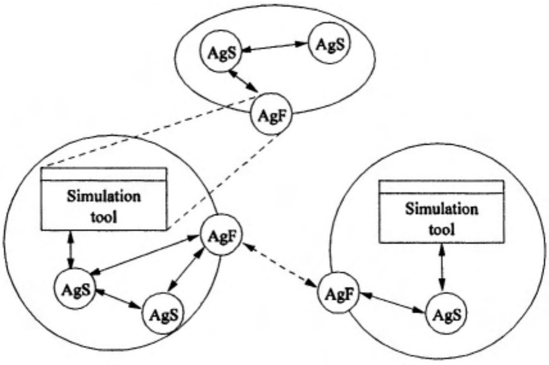


Figure 20-6. Multi-Agent Architecture allowing simulations ofindustrial systems

An infrastructure is proposed to allow a simulation process based on an agents' society (illustrated by Figure 20-6). It is mainly composed of interconnected agents. This principle permits to distinguish two classes of agents:

– the facilitators (AgF) facilitate the exchange of messages between simulation agents. They are intermediaries between agents' sub-societies. Moreover, the facilitators dynamically manage a knowledge database of the resources and the available services (resources, processing units, decisional centres...). They allow a better modularity and better dynamic's evolution support.

- the agents for simulation (AgS) are used during the simulation process for the decisional centres. The architecture is based on “white boxes” that must be filled with the behaviour defined in the abstract simulation model. This kind of agent is not entirely specified in MaMA-S. The proposed architectural skeleton is based on the AEIO’s facets [3]. This skeleton includes the interactions between the facilitators, and, in most of cases, between a facilitator and the environment’s objects (such as a simulation tool) [6,7].

Figure 20-6 illustrates another aspect of the MaMA-S approach: the recursive architecture. In fact, each agent or each environment object can be also a multi-agent system.

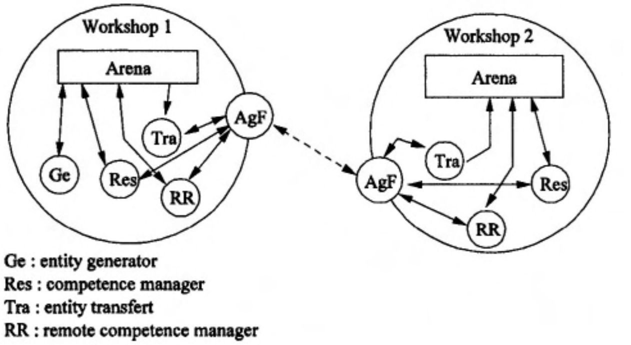


Figure 20-7. Example of the architecture of a multi-agent model

From the previous example shown for the specification, the following agent classes are highlighted:

- the agents corresponding to resource management decision centres,
- the agent corresponding to the entity generation centre,
- the agents that permits to send physical entities from a simulation model to another,
- the agents representing the “remote” resource managers.

Figure 20-7 illustrates the structure of the resulting multi-agent system. MaMA-S considers that agents are white boxes, which must be filled with the parts of the abstracts models. The means to interact between agents is proposed by MaMA-S (message syntax, service specification...).

2.4 Phase of Implementation

The implementation is the last phase which is adapted to the support of the modelling and the simulation of distributed industrial systems. Here, the objective is to translate the multi-agent model previously obtained into a computer model. Within this intention, this phase aims to choose the tools which will have to carry out the simulation (Arena®, Simple++®, QNAP...) and the multi-agent platform being used as support of the agent execution (SWARM, MadKit, Zeus, CORMAS, ARéVi...).

To tackle the translation of the multi-agent model, we propose a set of constraints that must be respect by the software. The major of them are:

- **for the simulation tools:**
  - The simulation tools must propose a communicating interface usable by the agents.
  - They must implement a set of behaviours whish are strictly equivalent to those awaited in the conceptual model.
  - The tools should not endanger the course of simulation. Thus, the simulation tool must be in conformity with the synchronisation policy of the models (the constraints of causality and vivacity must be respected).
- **for the agent platforms:**
  - The agents result directly from the conceptual model. Thus, the constraints of implementation are common to any multi-agent system (autonomy, interaction, distribution)

An implementation was proposed for the previous consortium example. It is based on the use of the simulation tool Arena®, and of Visual Basic® for the agents. Figure 20-7 also illustrates this choice of implementation.

Arena® permits to support the physical infrastructure of the system and to simulate the flow of physical entities inside this sub-system.

Visual Basic® is used to implement simple agents composed of a small communicating layer (based on sockets and message queues) and simple responding algorithms which correspond to the behaviours defined during the design. Moreover, those agents are temporally synchronized with a pessimistic approach.

3. CONCLUSION AND FUTURE WORKS

In this chapter, a methodological approach for the creation of simulation models of complex and distributed systems is proposed and presented. This approach, named MaMA-S, was previously specified in [6]. It facilitates the modelling and the simulation of decision-making processes, whether centralized or distributed. It also provides better reaction capacity in terms of modelling (systemic approach, etc.). Lastly, one of its strengths is its capacity to produce re-usable models.

Nevertheless, some applications (teaching application in the simulation scope and cyclic scheduling of a production system) enable to highlight certain weaknesses in our approach (see [6, 7] for more details). First of all, collaborative modelling is not fully taken into account by MaMA-S. Indeed, only a basic architecture for the collaborative support is proposed. In addition, some modelling elements need to be developed and proposed inside MaMA-S (dedicated template, etc.). They should allow an easier modelling. Moreover, this chapter presents the first works completed on MaMA-S. It still remains of many points for which a study proves to be necessary (synchronization of the simulation models, methodological guidelines...).

Finally, this methodological approach needs to move towards the standardization attempts that concern our areas of investigation: (i) the Unified Enterprise Modelling Language (UEML) for the modelling of distributed production systems, (ii) the Discrete-Event systems Specification (DEVS) to support distributed and interoperable simulations, (iii) the Foundation for Intelligent Physical Agents (FIPA) whose aims is to define the set of components for a multi-agent system platform. The previous points are currently being developed within the frame of an extension of the MaMA-S methodological approach.

REFERENCES

1. Banks J., 1999. Introduction to simulation. In: *Proceedings of the Winter Simulation Conference.* P.A. Farrington, H.B. Nembhard, D.T. Sturrock and G.W. Evans (Eds.), 7-13.
2. Butera F., 1991. La métamorphose de l'organisation : du château au réseau. Chapter of *Les nouveaux modèles d'entreprises : vers l'entreprise réseau.* Editions d'organisation, Paris, France.
3. Demazeau Y., 1997. Steps towards multi-agent oriented programming. In: *Proceedings of the 1st International Workshop on Multi-Agent Systems* (IWMAS), Boston, USA.
4. Dupont L., 1998. *La gestion industrielle.* Hermès, Paris, France.
5. Ferber J., 1995. *Les Systèmes Multi-Agents : Vers une intelligence collective.* InterEditions. Paris, France.
6. Galland S., 2001. *Approche multi-agents pour le conception et le construction d'un environnement de simulation en vue de l'évaluation des performances des ateliers multi-sites.* PhD thesis, Ecole Nationale des Mines, Saint-Etienne, France. December 2001. http://set.utbm.fr/membres/galland/recherche/publis/.
7. Galland S., Grimaud F., Beaune P. and Campagne J.P., 2003. MaMA-S: an introduction to a methodological approach for the simulation of distributed industrial systems. *International Journal of the Production Economics,* 85(1), 11-31. June 2003. http://set.utbm.fr/membres/galland/recherche/publis/.
8. Le Moigne J.L., 1992. *La modélisation des systèmes complexes.* Editions Dunod, Paris, France.

This page intentionally left blank

Index