THESIS FOR THE DEGREE OF DOCTOR IN LEADERS FOR TECHNOLOGICAL INDUSTRIES

Scheduling of Multipurpose Batch Plants

Towards the Development of a Decision-Making Tool for the Chemical-Pharmaceutical Industry

Samuel de Oliveira Moniz

Submitted to Faculdade de Engenharia da Universidade do Porto in partial fulfillment of the requirements for the degree of Doctor in Leaders for Technological Industries, supervised by Ana Paula Ferreira Dias Barbosa-Póvoa, Full Professor at Instituto Superior Técnico, and Jorge Manuel Pinho de Sousa, Associate Professor at Faculdade de Engenharia da Universidade do Porto.

Porto, March 2014

"A inquietante força que nos impulsiona rumo à incerteza e à descoberta transforma-nos em incansáveis e destemidos viajantes."

Rosa Gomes

This research was supported by the PhD Grant SFRH/BD/33970/2009, awarded by the Portuguese Foundation for Science and Technology, and by Hovione FarmaCiencia SA.

Agradecimentos

À professora Ana Paula Ferreira Dias Barbosa Póvoa, orientadora desta tese, pela forma como me acompanhou no decurso desta etapa, pela partilha de saberes, por toda a sua disponibilidade, confiança, determinação, simpatia e principalmente pelos incentivos constantes e tão valiosos para a concretização deste projeto.

Ao professor Jorge Pinho de Sousa, co-orientador desta investigação, pela transmissão de conhecimentos, inspiração e pela ajuda essencial no decurso desta etapa. A minha gratidão por todo o apoio e companheirismo que já se estende há 10 anos.

Ao Pedro Duarte, orientador do parceiro industrial (Hovione FarmaCiencia SA), pelo acompanhamento, colaboração e sobretudo pela visão inspiradora sempre presente.

Aos meus orientadores, agradeço o espírito único de trabalho em equipa que me permitiu chegar ao fim desta desafiante etapa.

À minha família por compreenderem a minha ausência e pelo incentivo incessante durante todo o percurso. Sem eles não teria sido possível.

Aos meus colegas e amigos do programa MIT Portugal pelo privilégio de trabalhar com eles, relembrando com especial carinho o tempo que passamos no Massachusetts Institute of Technology, pela forma como estimulou a investigação e as amizades que ficarão para sempre.

A todos aqueles que direta ou indiretamente contribuíram para a concretização desta tese, o meu sincero agradecimento!

Escalonamento de Instalações Multitarefa de Produção em Lotes

Desenvolvimento de uma Ferramenta de Suporte à Decisão para a Indústria Químico-Farmacêutica

Samuel Moniz

Resumo

Esta tese incide sobre o desenvolvimento de modelos de escalonamento para instalações multitarefa de produção em lotes que operam para a indústria farmacêutica. O problema de escalonamento da produção é geralmente reconhecido como um problema difícil de resolver uma vez que lida com vários objetivos potencialmente concorrentes.

A principal finalidade do escalonamento é produzir as quantidades adequadas, no tempo certo, com o menor custo e dentro dos critérios de qualidade. Para resolver este problema, modelos de otimização podem ser aplicados para obter soluções ótimas (ou quase ótimas). Os vários desafios que surgem a este nível estão relacionados com a implementação, modelação e eficiência computacional na resolução de problemas de grande dimensão. Contudo, a aplicação destes modelos em problemas reais cria claramente oportunidades de melhoria para as atividades de produção e logística.

Nesta tese é apresentada uma metodologia inovadora para a representação e resolução do problema de escalonamento, suportada por um modelo de otimização discreto. As características da indústria químico-farmacêutica levaram a uma definição mais alargada do problema de escalonamento, que tem em conta decisões relacionadas com a modificação dos equipamentos. Métodos de decomposição e estratégias de reformulação são propostas para abordar a complexidade computacional. A eficiência destes métodos é ilustrada através da resolução de instâncias reais. São também discutidos aspetos relacionados com a implementação da metodologia de escalonamento de forma a demostrar a sua aplicabilidade prática.

Scheduling of Multipurpose Batch Plants

Towards the Development of a Decision-Making Tool for the Chemical-Pharmaceutical Industry

Samuel Moniz

Abstract

The main objective of this thesis was to development scheduling models for multipurpose batch plants operating in the context of the pharmaceutical industry. The production scheduling problem is commonly recognized as being very difficult since it must deal with several potential conflicting objectives.

The primary goal of production scheduling is to produce the right amounts of product at the right time, cost, and quality. For that purpose, model-based approaches can be applied so as to obtain optimal (or close to optimal) scheduling solutions. Several challenges that arise at this level are related to implementation, modeling issues, and computational efficiency when solving large-scale problems. Nevertheless, the application of such models in real world scheduling problems clearly creates improvement opportunities for logistics and manufacturing activities.

In this thesis, an innovative methodology is introduced for efficiently representing and solving the integrated scheduling problem, based on a new general discrete-time model. The characteristics of the chemical-pharmaceutical industry led to the definition of an extended view of the scheduling problem that accounts for units redesign decisions. Decomposition methods and reformulation strategies are also introduced to address the computational complexity of the models. The effectiveness of the proposed methods is illustrated by solving several real world instances. Practical implementation issues of the scheduling methodology are also discussed so as to demonstrate its application potential.

Contents

1 Introduction

Industrial companies are continuously assessing their *Operations* with the objective of increasing the overall effectiveness of the manufacturing system. Markets where these organizations operate tend to become more complex over time, forcing companies to increase their responsiveness, both in terms of time and cost. The case of the pharmaceutical industry is a good example of how market is driving the change on product development cycles and manufacturing activities. Some of the most relevant driving factors are related to: a) the drought in new drug approval applications by the regulatory agencies such as the US Food and Drug Administration (FDA); b) the uncertainty associated to the Research and Development (R&D) and trials I-III phases; and c) the pressure on the drug prices and demand variability, caused by patent drops.

The cost for developing a new drug was on average $138 million in the 1970s and skyrocketed to $802 million by 1990, which represents an increase of 481% in capitalized costs (DiMasi et al., 2003; Hynes III, 2009). Recent data reveals that although the drug development cycle remained fairly stable (it can take as long as 15 years), the total cost of bringing a new drug to market is, in average, estimated to exceed $1 billion (Kessel, 2011). Nevertheless, the current worldwide paradigm imposes a reduction to less than 10 years from pre-clinical development to commercialization (Federsel, 2009).

This context is putting enormous pressure in the industry to reduce the time and the cost required to launch new drugs to market and, when drugs are in commercialization, reduce the manufacturing and inventory costs and the typically long production lead times. At the primary manufacturing (production of Active Pharmaceutical Ingredients (APIs)) and at the secondary manufacturing (formulation and packaging) it is not unusual for the overall supply chain cycle time to be 300 days. Moreover, the production of the APIs is considered the rate-limiting step of the supply chain (Shah, 2004).

Consequently, new technologies and production methodologies focused on the manufacturing system are being developed and evaluated, and some of them are being progressively adopted. For example, according to Roberge et al. (2005), 50% of the reaction tasks in the chemical-pharmaceutical industry could benefit from the adoption of continuous processes that have lower plant and production costs and can be highly automated. Concerning the manufacturing methodologies the focus is being on Good Manufacturing Practice (GMP), Process Analytical Technology (PAT) and also on advanced optimization tools, as stated by Grossmann (2005), with the concept of Enterprise-Wide Optimization (EWO).

The relevance of using optimization tools is being recognized by the industry (Klatt & Marquardt, 2009). Thus, not surprisingly, the efforts made in the past years by both academia and industry resulted into several successful integrations of optimization tools in complex decision-making processes related to process design, supply chain, planning and scheduling (Grossmann, 2005). Some relevant reviews on these topics have been published (Kallrath, 2005; Mendez et al., 2006; Barbosa-Povoa, 2007; Li & Ierapetritou, 2008; Maravelias & Sung, 2009; Verderame et al., 2010), showing the remarkable progress done in the Process System Engineering (PSE) area.

Production planning and scheduling are systematically considered very difficult functions to perform, since they are intended to produce operational plans dealing with several potential conflicting objectives, namely minimizing costs, completion times, and delays or maximizing profit. Additionally, these functions are closely related to other areas such as sales, procurement, production execution, and control, hence they may interface with decisions at the strategic and operational levels.

In short, we can say that the scheduling decision-making process must be simultaneously faster, integrated, validated, and provide various alternatives instead of a single solution to the problem. The assessment of the manufacturing system in new scenarios is done on a regular basis, and may be triggered by the arrival of new orders or rescheduling needs. Moreover, scheduling solutions must be obtained in reasonable time, considering the time window available for the decision-making process, and may need to consider decisions made in other areas. The validation of the solutions must then be performed to ensure operational feasibility, and since a variety of solutions having different objectives can be obtained, it is highly desirable to have alternative schedules.

The use of optimization methods in the scheduling decision making-process should ensure that all these requirements are met, thus providing a systematic way to obtain optimal (or close to optimal) scheduling solutions.

1.1 Research Gaps and Research Questions

The major issues raised by optimization approaches, when applied to industrial problems, are related to computational performance, uncertainty, multiscale optimization, or the modeling task itself (Grossmann, 2005; Grossmann, 2012). In particular, scheduling optimization is quite difficult to perform since it involves solving combinatorial problems in highly collaborative and dynamic production environments. The relevance of these issues and the practical need to address them, in an effective and efficient way, are the main motivation for this work. Our research is in fact pursued with the aim of designing new optimization based decision-support tools, for solving planning and scheduling problems in process batch plants. Thus, in spite of the significant progress done in the development of scheduling models, these issues are restricting the adoption of mathematical approaches by the process industry. Modeling is surely a very critical issue, since it deals with the design of models targeting their integration with the company's decision-making processes. Thus, the first research question can be posed as follows:

Q1: *What should be the structure and components of models for scheduling multipurpose batch plants?*

Question Q1 reflects the importance of understanding the structure of the scheduling problem, as seen by the practitioners, and leads to the definition of the main components that will constitute the decision-making tool. Moreover, the factors that are determinant to integrate optimization models in industrial practices should also be considered here. In other words, it is quite relevant to understand the structure of the scheduling problem in order to integrate models in existing decision-making processes, in such a way that models can be rapidly understood by the industrial practitioners. Along the same lines, another motivating research question arises:

Q2: *What is the optimal strategy for scheduling of multipurpose batch plants taking into account the different production modes?*

Question Q2 provides guidance to this research on the development of scheduling models and solution methods, capable to solve large-scale scheduling problems and to adapt to different production scenarios. The critical factors concern with the computational performance and quality of the solutions when solving difficult scheduling problems. The challenge here is to develop solution methods such that the balance between computational time and quality of the solutions is acceptable in the context of the day to day scheduling decisions.

1.2 Thesis Objectives

The main objective of this thesis is to develop a general methodology for solving scheduling problems of multipurpose batch plants. In a more detailed way, this work aims at:

1) identifying the requirements for the scheduling models;
2) developing generic scheduling models for multipurpose batch plants;
3) designing solution methods;
4) developing the basis for a decision support tool for the chemical-pharmaceutical industry.

First we try to develop a clear view of the scheduling problem as seen by the industry, and to identify the requirements necessary to design scheduling models. Then we design, test, and validate scheduling models under real-world conditions, which can be supported by the case study addressed in this work. We then propose solution methods to solve large-scale scheduling problems. Finally, we develop the prototype of a decision-support tool that integrates the developed mathematical approaches with decision-making processes. The characteristics of the models, the structure of the problems and also the decision-making processes of the case-study are taken into consideration, so as to define a scheduling methodology that can be truly implemented in real manufacturing systems.

1.3 Scope of the Thesis

The work presented throughout this thesis is centered on the scheduling problem of multipurpose batch plants and is motivated by the resolution of a case study from the chemical-pharmaceutical industry. The level of analysis of this research is the multipurpose batch plant, and the unit of analysis is the scheduling model.

The case-study of this work was designed based on a company that is responsible for the development and manufacturing of complex chemicals called Active Pharmaceutical Ingredients (API). The batch processes are typically long and are executed under close supervision of the regulatory authorities. To manufacture a single product, several days of effective production time may be required, with tasks processing times varying between one hour and two days. The production resources are shared between products that are under development and products that are already in commercialization. Changeovers are required to avoid cross contamination of the products and are quite critical since often they impose significant downtime periods. In this context, the scheduling problem consists in efficiently allocating production resources to tasks so as to fulfill given demand targets. In the cases when the scheduling problem is deeply dependent on other types of problems, such as batch plant design or planning, those problems have also been considered.

Two important advantages of this study should be emphasized. First, the resolution of real world scheduling problems helped to focus the research in the development of optimization models that can effectively be implemented. Thus, practical scheduling requirements have been discussed with the company and considered in the models whenever possible. Second, although the scheduling models and solution approaches developed were motivated by a case-study, they can be applied to other types of industries, as long as the scheduling problem has a similar structure.

1.4 Key Concepts and Definitions

The main key concepts and definitions used throughout this thesis are:

a) Production planning – *"Production planning is viewed here as the planning of the acquisition of the resources and raw materials, as well as the planning of the production activities, required to transform raw materials into finished products* *meeting customer demand in the most efficient or economical way possible.”*, by Pochet and Wolsey (2006).

b) Production scheduling – “*Scheduling is a decision-making process that is used on a regular basis in many manufacturing and services industries. It deals with the allocation of resources to tasks over given time periods and its goal is to optimize one or more objectives*”, by Pinedo (2002).

c) Multipurpose batch plants – *“multipurpose batch plants or jobshops are general purpose facilities where a variety of products can be produced by sharing the available equipment, raw materials and intermediates, utilities and production time resources”*, by Barbosa-Povoa (2007).

Planning is associated to long-term decisions and scheduling is related to short-time decisions. The interaction between these two types of decisions has been extensively addressed in the literature. A recent review on this topic is provided by Maravelias and Sung (2009).

1.5 Research Design and Methods

The models developed in this thesis apply well known mathematical approaches, such as Linear Programming (LP) and Mixed Integer Linear Programming (MILP). To address the computational complexity of some scheduling instances, decomposition methods have also been used.

Modeling scheduling problems invariably requires integer variables, which results into models having both continuous and integer variables. Due to the combinatorial nature of the problems, a complete enumeration of all possible values of the decision variables is impractical. To solve this type of problems the branch-and-bound (B&B) technique is generally applied. B&B is an enumeration algorithm that applies a partitioning process to cut a lot of the enumeration whenever possible. Current implementations of B&B take advantage of extraordinary theoretical progresses and are in practice quite efficient.

In a nutshell, B&B computes the *Linear Relaxation* of the MILP problem at each node of the enumeration tree and keeps record of the best integer solution found and of the linear relaxation solution and value to find bounds on the optimal values for the integer programs (Johnson et al., 2000). Nowadays, B&B is part of advanced solvers such as CPLEX that include a preprocessing stage and heuristics to speed up the resolution time.

However, many scheduling problems can hardly be solved by exact methods (as B&B) in an acceptable amount of time. These problems usually deal with a large number of resources, tasks, and mainly with a large number of time intervals. Moreover, if we consider some complicating constraints as sequence-dependent changeovers and temporary storage in the processing units, the resolution time tends to be prohibitive and the quality of the solutions tends to deteriorate very rapidly. Alternatively, decomposition techniques can be applied to obtain satisfactory solutions quickly. In this work, we have applied aggregated model formulations, task-unit aggregation, and time-based decompositions.

In this research, the case study naturally played a very important role. Data collection for analysis and interpretation was performed in the company and used later to build the scheduling instances. Due to the lack of coherent information structures, data was firstly arranged to be used then by the scheduling models. In order to test and validate the proposed methodology, we have, during one year, performed meetings in a regular basis with process engineers and planners. Insights from industrial practitioners revealed to be very useful in redefining the components of the methodology and the integration requirements between those components.

1.6 Thesis Outline

The research questions have been tackled throughout four papers that essentially constitute the body of the thesis, thus some unavoidable repetitions are present in this document. The original content of each published or submitted paper was transcribed to single chapters, excluding the work presented in Chapter 2 that was not submitted to a journal. Thus, the relevant literature review is discussed in each of these chapters.

Chapter 2 presents a discussion on the complexity of planning and scheduling functions. The role and scope of planning and scheduling are addressed in detail, in an attempt to better understand the critical factors that drive these functions in the context of the pharmaceutical industry. We show that there is an improvement path that must be followed in order to respond to the new challenges of this industry.

Chapter 3 introduces a new type of problem that is closely related to the scheduling problem. The equipment redesign problem is defined by the implementation of modifications in the processing units so as to change their suitability to perform tasks. This problem takes more relevance in industries that perform process development, since the production recipes evolve with it and for that reason it may be necessary to modify the processing units. Modeling both problems simultaneously increases the solution space, since additional task-unit assignments can be explored by modifying the processing units. The developed model delivers a schedule and an equipment modification plan.

Chapter 4 proposes a new general discrete-time scheduling model for multipurpose batch plants. The developed formulation deals efficiently with two complicated requirements of the discrete-time scheduling models as: the sequence-dependent changeovers and the temporary storage in the processing units. Other operational requirements such lots blending and material flows traceability that have been somehow neglected by the literature, are also taken into account.

Chapter 5 presents a novel solution methodology for the production scheduling of batch plants that distinctively integrates the representation of the scheduling problem, the optimization model, and the decision-making process. The main objective is to ensure the development of a methodology that can effectively be integrated in the decision-processes of the company, in which the case-study of this research project is based.

Chapter 6 addresses the scheduling of regular and non-regular production. The objective of this chapter is two-fold. First, solving a scheduling problem that requires two distinct operating strategies: campaign and short-term production. Second, tackling the computational complexity of the scheduling problems by using decomposition methods and reformulation strategies. Real world scheduling instances are solved to demonstrate the efficiency and quality of the solutions.

Finally, chapter 7 summarizes the work presented in this thesis and identifies some future research topics in the area.

The papers that support the structure of the thesis, described above, are presented in Table 1.1 and have been published/or are under review in international peer reviewed journals or have been published in conference proceedings.

Table 1.1 – Publications.

Chapter	Publications
Chapter 3	Moniz, S., Barbosa-Póvoa, A. P., & Pinho de Sousa, J. (2012). Scheduling with equipment redesign in multipurpose batch plants. Foundations of Computer-Aided Process Operations - FOCAPO 2012.
Chapter 4	Moniz, S., Barbosa Póvoa, A. P., & Pinho de Sousa, J. (2013). A new general discrete-time scheduling model for multipurpose batch plants. Industrial & engineering chemistry research. doi: 10.1021/ie4021073.
Chapter 5	Moniz, S., Barbosa-Póvoa, A. P., & Pinho de Sousa, J. (2013). A solution methodology for scheduling problems in batch plants. *Under review in* Industrial & engineering chemistry research.
Chapter 6	Moniz, S., Barbosa-Póvoa, A. P., & Pinho de Sousa, J. (2013). Simultaneous regular and non-regular production scheduling of multipurpose batch plants: a real chemical-pharmaceutical case study. *Under review in* Computers & Chemical Engineering.

Papers in Conference Proceedings

Moniz, S., Barbosa-Póvoa, A. P., & Pinho de Sousa, J. (2012). Regular and non-regular production scheduling of multipurpose batch plants. Proceedings of the 22nd European Symposium on Computer Aided Process Engineering. doi: 10.1016/B978-0-444-59520-1.50012-9.

Moniz, S., Barbosa-Póvoa, A. P., & Pinho de Sousa, J. (2013). Extending the Resource-Task Network (RTN) for Industrial Scheduling Problems. IO 2013, XVI Congresso da Associação Portuguesa de Investigação Operacional.

References

Barbosa-Povoa, A. P. (2007). A critical review on the design and retrofit of batch plants. Computers & Chemical Engineering, 31, 833-855.

DiMasi, J. A., Hansen, R. W., & Grabowski, H. G. (2003). The price of innovation: new estimates of drug development costs. Journal of health economics, 22, 151-185.

Federsel, H.-J. r. (2009). Chemical Process Research and Development in the 21st Century: Challenges, Strategies, and Solutions from a Pharmaceutical Industry Perspective. Accounts of Chemical Research, 42, 671-680.

Grossmann, I. (2005). Enterprise-wide optimization: A new frontier in process systems engineering. AIChE Journal, 51, 1846-1857.

Grossmann, I. (2012). Advances in mathematical programming models for enterprise-wide optimization. Computers & Chemical Engineering.

Hynes III, M. D. (2009). Project and capacity management: An application to drug development. Computers & Chemical Engineering, 33, 1994-1998.

Johnson, E. L., Nemhauser, G. L., & Savelsbergh, M. W. P. (2000). Progress in linear programming-based algorithms for integer programming: An exposition. INFORMS Journal on computing, 12, 2-23.

Kallrath, J. (2005). Solving planning and design problems in the process industry using mixed integer and global optimization. Annals of Operations Research, 140, 339-373.

Kessel, M. (2011). The problems with today's pharmaceutical business [mdash] an outsider's view. Nature biotechnology, 29, 27-33.

Klatt, K.-U., & Marquardt, W. (2009). Perspectives for process systems engineering—Personal views from academia and industry. Computers & Chemical Engineering, 33, 536-550.

Li, Z., & Ierapetritou, M. (2008). Process scheduling under uncertainty: Review and challenges. Computers & Chemical Engineering, 32, 715-727.

Maravelias, C. T., & Sung, C. (2009). Integration of production planning and scheduling: Overview, challenges and opportunities. Computers & Chemical Engineering, 33, 1919-1930.

Mendez, C. A., Cerda, J., Grossmann, I. E., Harjunkoski, I., & Fahl, M. (2006). State-of-the-art review of optimization methods for short-term scheduling of batch processes. Computers & Chemical Engineering, 30, 913-946.

Pinedo, M. (2002). Scheduling : theory, algorithms, and systems. Upper Saddle, N.J.: Prentice Hall.

Pochet, Y., & Wolsey, L. A. (2006). Production planning by mixed integer programming: Springer.

Roberge, D. M., Ducry, L., Bieler, N., Cretton, P., & Zimmermann, B. (2005). Microreactor technology: a revolution for the fine chemical and pharmaceutical industries? Chemical engineering & technology, 28, 318-323.

Shah, N. (2004). Pharmaceutical supply chains: key issues and strategies for optimisation. Computers & Chemical Engineering, 28, 929-941.

Verderame, P. M., Elia, J. A., Li, J., & Floudas, C. A. (2010). Planning and Scheduling under Uncertainty: A Review Across Multiple Sectors. Industrial & engineering chemistry research, 49, 3993-4017.

2 On the Complexity of Production Planning and Scheduling in the Pharmaceutical Industry

Abstract

This chapter discusses on the role of the planning and the scheduling functions in the drug development process and production environment of the pharmaceutical industry, and aims at identifying the critical factors of decision-making and global optimization to the operations. We redefine the scope of planning and scheduling problems, and we propose an extended view of these problems to account for higher levels of integration between process design and operations. Finally, we introduce a conceptual representation, the Delivery Tradeoffs Matrix to provide guidance on the tradeoffs occurring in the drug development process and to expose the factors that affect the performance of these manufacturing systems.

Keywords: process design; planning and scheduling optimization; batch plants; pharmaceutical industry

2.1 Introduction

The pharmaceutical industry can be view as a complex system of processes, operations and organizations involved in the discovery, development and manufacturing of drugs (Shah, 2004). Companies operating in this industry are responsible for: research and development (R&D) activities; development and manufacturing of active pharmaceutical ingredients (APIs); and drugs manufacturing. From the supply chain perspective, different companies collaborate in the development and manufacturing activities as a way to reduce the risks involved in the long product development cycle. This collaboration is characterized by technological information sharing and operations synchronization so as to ensure time-to-market of new drugs.

The product development cycle includes pre-clinical research, clinical studies on humans (trials I-III) and commercialization phases, and involves several fields such as process chemistry, analytical chemistry, process engineering, process safety, regulatory compliance and plant operation that must be effectively applied (Federsel, 2009). The same author states that although the pharmaceutical industry has historically tolerated total time investments of more than 10 years from idea to market, the current worldwide paradigm imposes a reduction of this time. The launch of new drugs in the market involves the development of new substances for specific treatments, and their manufacturing in substantial quantities to satisfy the demand trough costs effective operations. Here, the pharmaceutical industry is confronted with several challenges that are related to increasing R&D costs, long cycle times and low probabilities of success (Hynes III, 2009).

In general, manufacturers and regulators create a specific context to the operations management, thus conditioning planning and scheduling functions. The planning problem involves the determination of strategic production plans, in which decisions are typically made assuming a certain degree of aggregation of resources and time, hence defining bounds to the scheduling problem. The scheduling problem involves the determination of operational plans at the level of the most elementary production resources and at a fine time grid. Both problems present potentially several conflicting objectives such as, for example, minimizing costs and delivery times. The characteristics of the market, of the production processes and the chemical plants make planning and scheduling tasks particular difficult to perform. We have looked into the literature addressing planning and scheduling problems in the pharmaceutical industry and tried to derive conclusions concerning their scope and challenges.

The rest of the paper is structured as follows. Section 2.2 introduces a broad problem description for discussing the scope of planning and scheduling decision-making. Section 2.3 presents critical factors that determine how planning and scheduling is done in the pharmaceutical industry. In section 2.4, the Delivery Tradeoffs Matrix is introduced. Finally, in section 2.5, some concluding remarks are presented.

2.2 The Scope of Planning and Scheduling

Planning and scheduling refer to procedures of allocating resources to execute chemical and physical processing tasks (Reklaitis, 1992). Planning is typically associated to long-term horizons, while scheduling is related to short-time horizons. The time horizon and level of detail of planning and scheduling decisions are usually not fixed a priori since they depend on the specific problem being addressed. Moreover, planning and scheduling integration is also case specific, depending mainly on the types of decisions performed and desirable aggregation/detail at each decision-making level.

Planning and scheduling are deeply dependent on other corporate functions such as sales, procurement and production execution and control. See for example the process operations hierarchy of Bassett et al. (1996), in which planning and scheduling are part of a hierarchical and bi-directional process involving several functions, at tactical and strategic decision-making levels.

In the context of the process development and manufacturing of drugs, R&D and Operations Management (OM) departments perform critical activities that determine how planning and scheduling are effectively done. A visualization of these activities is presented in Figure 2.1. The first step, *Process Synthesis*, refers to the quantitative specification of physicochemical materials manipulations that take place, having as output a recipe that is independent of particular processing units. In other words, the recipe describes the chemistry steps required to manufacture the product. After the chemical process has been validated in laboratory it follows the *Process Scale-up*. This step complies with the development of the chemical process so as to pass from a laboratory scale to an industrial production dimension, resulting in the determination of the final product quantities (lot sizes) and an initial assessment of the processing times.

Note that several process scale-ups are typically performed to respond to the demands at the early stages of the development, at the clinical trials I-III and also after the drug approval for commercialization.

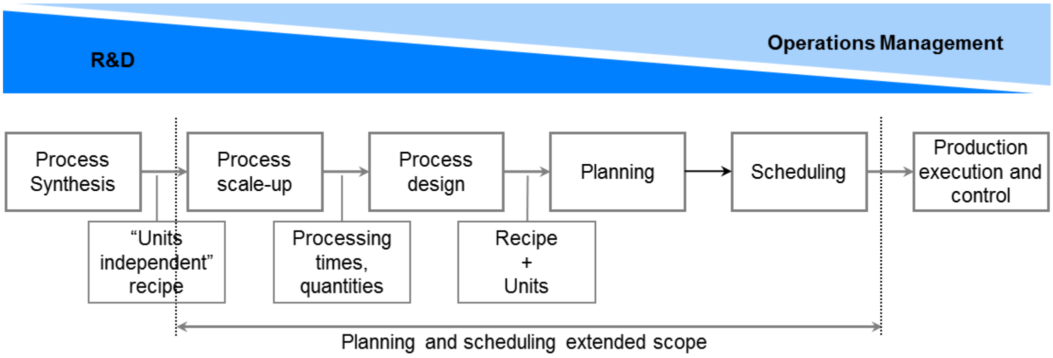


Figure 2.1 – Scope of the planning and scheduling problems.

The next step, *Process Design*, consists in using the information available to develop an industrial process. Here, the characteristics of the products and processing units are considered to seek the development of an efficient production process concerning resources utilization, given a set of market and operating constraints. In particular, the network of processing tasks is analyzed and the suitable processing units are determined. The complexity increases with the following two steps. *Planning* and *Scheduling* need to deal with the utilization of the production resources in the most efficient manner, and must account for the uncertainty associated to parameters such as tasks processing time or demand. The final step, *Production Execution and Control* involve the following activities production dispatching, control actions and quality assessment, among others.

Typically, the overall decision process is assigned to the R&D and the OM departments. The first steps are mainly associated to the R&D functions, while the OM deals essentially with planning and manufacturing. Nevertheless, decisions should be performed collaboratively in order to ensure that decisions made at each department are properly considered. Although Figure 2.1 suggests a sequential and directional decision flow, the different steps are often overlapped and revisited whenever necessary.

We argue that planning and scheduling functions are extended in order to integrate some decisions made in the process scale-up and design steps. The planning problem, either of long-term or short-term, benefit from considering decisions taken at the scale-up and process design levels, since these decisions have a direct impact on the determination of the processing units suitable for the process, resulting into different production routes (alternative processes). On the contrary, after schedule release to the shop-floor, changes on planning and scheduling decisions are very limited, although rescheduling is a common practice. The same happens with changes in process design decisions that may not be possible or are not desirable to perform. On the same lime, Barbosa-Povoa and Macchietto (1994) state that design and scheduling aspects must be considered simultaneously. The design of batch plants has also been object of an extensive review by Barbosa-Povoa (2007).

In summary, we argue that the scope of the planning and scheduling functions must be extended to account for design decisions, especially for manufacturing chemical processes that are under development. This will increase the solution space of planning and scheduling decisions, targeting the global optimization of the operations. Moreover, in the case of the pharmaceutical industry, planning and scheduling are determined by specific drivers that we group in three categories: *i*) market; *ii*) processes; and *iii*) plants, all of them having impact on the decision-making process depicted in Figure 2.1. These drivers are briefly discussed in the following section.

2.3 Critical Factors

Planning and scheduling are functions that aim primarily at reducing costs and improving responsiveness of the manufacturing systems. The critical factors that drive planning and scheduling functions, in the particular context of the pharmaceutical industry, can be grouped in three categories: *Market, Processes* and *Plants*. Market factors are related to the contextual factors specific of this industry. Process factors have to do with the structure of the chemical processes. Plant factors concern to the operating strategies and resources characteristics of the manufacturing systems. It is important to note that some of the process and plant characteristics discussed in the following subsections are not specific of the pharmaceutical industry.

2.3.1 Market

The market context has a direct influence on planning and scheduling functions. First of all, the pharmaceutical market is highly fragmented (Basu et al., 2008). There is a large variability on the demand, also as a result of the pressure created by generic drugs, which leads to a larger production mix in the manufacturing sites. Operations flexibility is therefore required to fit the system to this demand, and for that, efficient planning and scheduling methods are required.

Regulatory agencies such as the US Food and Drug Administration (FDA) or the European Medicines Agency (EMA) impose strict regulations that go from the development to the manufacturing of drugs. Manufacturing in a high regulated market has to deal with additional complexities that do not exist in less regulated markets. Chemical processes are executed under a close supervision of the regulatory agencies that define procedures to monitor process changes. For example, in the manufacturing of APIs, a validated and certified production process can have its lot size only vary up to a maximum of 10%. To change more than that, the process has to undergo a new certification process, which will increase costs and require more non-production time.

Moreover, scheduling and planning must account for constraints imposed by long product development cycles. The development process intrinsically defines the set of production resources (processing units) that can be used in each phase, thus conditioning planning and scheduling decisions.

Globally, the time-to-market issue and pressure to reduce costs are imposing operations to run more efficiently and therefore advanced planning and scheduling methods are necessary (Moniz et al., 2013 *submitted*).

2.3.2 Processes

The process topology determines the scheduling models that can be applied. Processes can be classified as *Sequential* and *Network* process. In short, sequential processes do not allow batch mixing and splitting, thus the batch entity is preserved. Network processes have arbitrary networks of processing tasks and batch mixing and splitting is allowed. Comprehensive reviews on the classification of the batch scheduling problems are available in Pinto and Grossmann (1998) and in Mendez et al. (2006). One relevant aspect concerning modeling of scheduling problems is that the process topology (sequential or network) is determined by material handling constraints and not by the plant structure (*e.g.*, processing units, connectivity) (Sundaramoorthy & Maravelias, 2011; Maravelias, 2012).

Sequential processes may have similar tasks sequences, thus may be executed in a *Multistage* facility, or on the other hand, may have product specific tasks sequences and require a *Multipurpose* facility. Additionally, there are network processes that have arbitrary structures and also require a multipurpose facility. In practical terms, modeling planning and scheduling decisions may be quite different if the plant is a multistage or multipurpose. For example, planning and scheduling of multistage batch plants can be focused on the bottleneck stage, since this stage can often be identified. On the contrary, in multipurpose plants the bottleneck units tend to change with the production mix, which requires the adoption of different decomposition approaches.

Processes can have batch, semi-continuous and continuous tasks and produce materials subject to different storage policies, such as *Unlimited Intermediate Storage* (UIS), *Finite Intermediate Storage* (FIS), *Zero-Wait* (ZW) and *Non Intermediate Storage* (NIS).

In the manufacturing of APIs, for example, processes require numerous production steps with tasks having short and long processing times, which may span across several working shifts. Regulatory and quality procedures define the lot size and changeover requirements that must be rigorously followed in the manufacturing sites, thus introducing additional time to the effective production time. Stable intermediaries and final products are produced in lots, and therefore lots traceability must be ensured. Thus, proportions/quantities of each lot used in subsequent lots must be recorded. Changeovers are needed to avoid cross-contamination of the products and have the immediate consequence of increasing the idle time of the processing units. Moreover, the cleaning times of units are typically shorter when changing the production to lots of the same product, and are usually larger when changing to a different product.

The first batches after a scale-up are usually more difficult to execute, since it may involve using different processing units or even perform changes in the process. Additionally, if the process is under development, it is more difficult to perform scheduling, since at this point the knowledge about the process is very limited. For that reasons, processes impose frequently the revision of the schedule.

2.3.3 Plants

The plant structure has also implications on how planning and scheduling is performed. Note that, although the modeling approach strongly depends on the process topology (as discussed above), the characteristics of the plants (such as resources, plant structure, operating mode, and batch/continuous manufacturing) lead as well to specific planning and scheduling problems.

Batch plants have different types of production resources (*e.g.*, processing units, storage units, units' connections, materials, utilities and people) that may need to be considered when solving these problems. Facilities having multipurpose units are inherently more flexible, since these units are suitable to produce a variety of products (Barbosa Povoa & Macchietto, 1994). Resources of the type material may be available for manufacturing several processes according to the recipe instructions, and they can have the following states: raw materials, intermediaries, stable intermediaries and final products. Material storage policies are defined by the process itself and by the storage alternatives available for scheduling. Dedicated storage units may exist, even if multipurpose units such as reactors can often be used temporarily as vessels.

The plant is typically composed by reactors, filters and dryers that are connected through a complex system of pipelines or through mobile vessels. This connectivity allows fixed or flexible links between units, having significant implications on the effective utilization of processing units. In general, resources sharing and connectivity alternatives are advantageous in scenarios were product demands or formulations demands change rapidly (Barbosa Povoa & Macchietto, 1994). Design and scheduling problems have been addressed simultaneously as a way to account for units connectivity and layout characteristics, resulting into integrated operating strategies (Barbosa-Povoa, 2007). Barbosa Povoa and Macchietto (1994) addressed for the first time the design and scheduling of multipurpose batch plants taking into account the plant structure and, more recently, the units redesign problem was introduced by Moniz et al. (2012). Here, we have considered that the units' suitability to perform tasks can change during the scheduling horizon. The set of resources available in the plant and the degree of flexibility to adapt the resources to the products demand will provide more or less scheduling alternatives.

Concerning the operating mode, batch plants that have to manufacture products with dissimilar recipes and low production volume require different operating strategies than those having similar recipes and high volume (Reklaitis, 1992). In this way, for low volume production (*short-term mode*) a reduced number of batches are produced and for high volume production (*campaign mode*) the number and size of batches tends to be higher. These strategies have impact on the way how resources are allocated and on the required system responsiveness. The short-term mode under a multipurpose environment requires a higher responsiveness from the manufacturing system, since resources are shared between several products in a very dynamic production environment, whereas in the campaign mode, resources are allocated to single products during long time periods. Campaign schedules can be computed using the periodic scheduling approach proposed by Shah et al. (1993) where it is assumed that tasks are executed with a cyclic pattern. In practice, these schedules are seen as operationally easier to manage and execute. Finally, we may have a mixed strategy, in which some resources are allocated to short-term demand, while the other resources are dedicated to the campaign demand (Moniz et al., 2013).

Continuous manufacturing of pharmaceuticals is an emergent process mode that relies on flow reactors and is being evaluated to the production of drugs. An immediate consequence of using flow reactors, instead of using batch reactors, is that the production process moves from a batch process to continuous operating conditions (Buchholz, 2010). Benefits of continuous manufacturing when compared to batch manufacturing include lower plant and production costs, lower carbon footprint; better quality, higher safety; less costs to scale-up, and higher levels of automation (Roberge et al., 2008; Calabrese & Pissavini, 2011). Nevertheless, existing technological challenges of flow reactors and adaptation of batch processes to continuous processes have made their evaluation and deployment difficult.

I what concerns the supply chain of continuous processes, materials can be planned in a regular basis so leading to a reduction of inventory costs. Moreover, since labor costs in the pharmaceutical industry are very significant (Roberge et al., 2008), the possibility of introducing automated continuous processes has at least two advantages: it reduces labor costs and improves the reliability of the production process, thus reduces the uncertainty in planning and scheduling problems.

2.4 Delivery Tradeoffs Matrix

The ultimate goal of planning and scheduling is to deliver the right amounts of product at the right time, cost and quality. Thus, in order to provide guidance on the issues that determine the effectiveness of launching a new drug to the market, we propose a conceptual representation, named the *Delivery Tradeoffs Matrix* (DTM) depicted in Figure 2.2. The relative importance of costs and uncertainty on the manufacturing activities that support the development and delivery of APIs or final products can be assessed in the DTM.

Drug Development Cycle

The matrix depicts three phases (R&D, trials I-III and commercialization) of the drug development cycle. The R&D phase accounts for discovery, safety and toxicology research activities, and clinical supplies. Trials I-III are related to the clinical studies performed on humans. The commercialization phase includes the manufacturing activities required to deliver the right amounts of product to the market, after approval by the regulatory agencies. Uncertainty and costs are represented in a scale of high-low and the lot size proportion at each phase is indicated by the size of the associated bubble. The DTM of Figure 2.2 a) attempts to show the current tradeoffs of the industry, while b) tries to depict a future scenario, as a possible response to the challenges the pharmaceutical industry is facing and needs to overcome.

The DTMs were built taking into account a set of estimated values available in the literature. To the best of our knowledge, there are no reliable figures regarding the cost structure and uncertainty associated to the R&D, trials I-III and commercialization phases (Suresh & Basu, 2008). Though, the dimension of development and manufacturing costs justify a discussion on the path to the manufacturing efficiency of the pharmaceutical industry.

Uncertainty and Costs

At the start of a research program, products and processes are not developed, and therefore there is a high uncertainty associated to the drug structure and to the process design. Uncertainty makes planning decisions more complex, since it is more difficult to estimate the required time and resources. For example, in the development and manufacturing of APIs it is common to allocate production resources 6 to 12 months in advance. Thus, any changes in the planning will surely have impact in manufacturing costs and delivery time.

With drug development the uncertainty tends to decrease since product and process characteristics are better understood. At the laboratory scale just small amounts are produced (around few hundred grams). The delivery of the first scaled up batch (usually between 1 to 5 kg), used to support toxicological and formulations studies and phase I trials, is on the critical path of the development process This scale-up is particular difficult to perform since the knowledge from the laboratory scale is typically not sufficient to guarantee a successful process at a plant scale (Federsel, 2009). Moreover, the drug development process requires a series of scale-ups so as to develop an efficient production process. At the commercialization stage, there is an increasing need for API or drug product at the order of hundreds of kilograms. The processes are well defined, thus the uncertainty is mainly associated to market parameters such as demand and to the processing time of complex production tasks.

The current practice demonstrates that there are large costs and high uncertainty at the R&D and trials I-III phases (see Figure 2.2). The total cost of bringing a new drug to market is estimated to exceed 1 billion dollars (Kessel, 2011). In terms of the total cost structure, pharmaceutical R&D costs are around 30% to 35% and clinical trials (typically representing the most significant cost) can be between 35% to 40% of the total (Suresh & Basu, 2008).

Time-to-Market and Amount Delivered

It should be noted that from the planning and scheduling perspective, the delivery of products to Trials I-III phases is of extreme importance. Shah (2004) and Buchholz (2010) pointed out that time-to-market is a critical driver of the pharmaceutical industry. Additionally, Buchholz (2010) highlighted another relevant driver for this industry, which is fast and robust scalability of the production processes. These drivers are even more relevant since frequently more than one company are developing drugs targeting the same market, thus the importance to respect due dates is crucial.

On the other hand, at the commercialization phase there is more flexibility concerning delivery dates, if there is inventory on the supply chain. According to Shah (2004), the whole pharmaceutical chain stock can represent 30% to 90% of the annual demand in quantity. Therefore, at this phase, we can say that delivering the right product amounts is relatively more important than respecting delivery dates. Note that the lot sizes at the Trials I-III phases are in the order of few kilograms, while after several scale-up and validation steps, the lot sizes are around hundreds of kilograms. After drug development, the manufacturing costs are lower and tend to decrease with the reduction of the root causes of variability in the process.

Concerning the operating mode, manufacturing sites run in short-term mode to fulfill a small product demand, or run preferably in campaign mode to respond to a regular demand. The short-term mode is also used for manufacturing products that are in commercialization, this naturally resulting in the production of a smaller number of lots. However, in all cases the process must run with the same lot size as approved by the regulatory agencies.

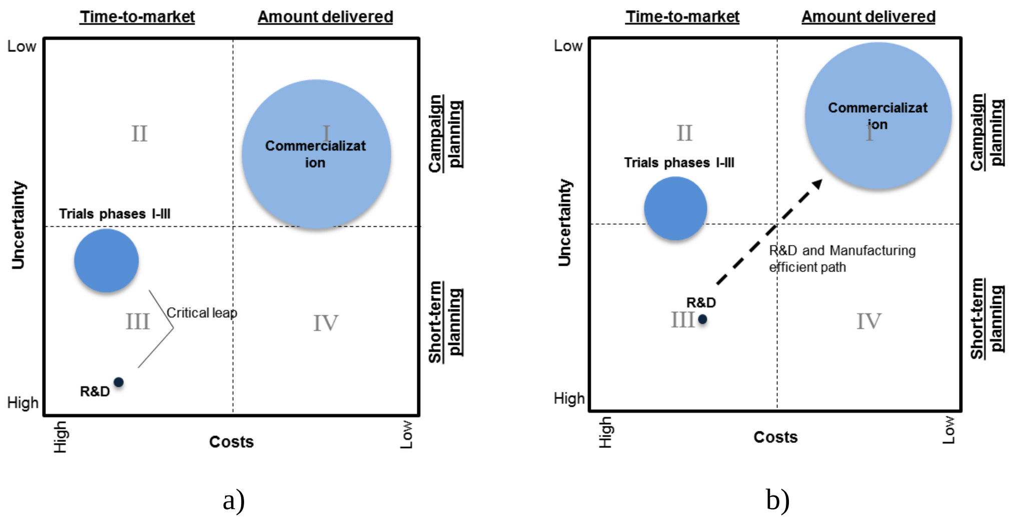


Figure 2.2 – Delivery Tradeoffs Matrix (DTM) of the pharmaceutical industry: a) current state, b) future scenario.

The Path to Efficient R&D and Manufacturing Activities

All these issues led the pharmaceutical industry to recognize the need for reducing time-to-market, the costs of new drug development, and manufacturing costs. The path to efficient R&D and manufacturing activities has to find ways to address uncertainty and reduce costs, see Figure 2.2 b). This will involve the introduction of new production technologies (Suresh & Basu, 2008), as well as, the adoption of innovative process design and planning and scheduling decision-making tools (Shah, 2004). For example, according to Roberge et al. (2005), 50% of the reaction tasks in the chemical-pharmaceutical industry could benefit from the adoption of continuous processes based on the microreactor technology. In what concerns decision-making, the relevance of applying optimization methods and deploying more integrated decision-making processes is being recognized by the industry, despite the challenges that still exist (Grossmann, 2012).

Figure 2.2 b) provides a view on a possible path for efficient R&D and operations. The uncertainty and costs can be reduced by the adoption of new technologies (e.g., continuous flow manufacturing, process analytical technology (PAT)) and, on the other hand, be addressed by optimization tools.

2.5 Final Remarks

This piece of research is intended to analyze the main aspects that influence planning and scheduling decisions in the context of the pharmaceutically industry. Extending the traditional scope of planning and scheduling functions is particularly interesting, if drug development and manufacturing activities are simultaneously considered.

The critical factors that determine planning and scheduling were identified and grouped in three categories: market, processes and plants. In our view, comprehensive optimization methods for the pharmaceutical industry must somehow take into account these factors.

Finally, we propose a conceptual representation, the Delivery Tradeoffs Matrix that attempts at providing guidance on uncertainty and costs issues involved in the drug development and manufacturing activities.

References

Barbosa-Povoa, A. P. (2007). A critical review on the design and retrofit of batch plants. Computers & Chemical Engineering, 31, 833-855.

Barbosa-Povoa, A. P., & Macchietto, S. (1994). Detailed design of multipurpose batch plants. Computers & Chemical Engineering, 18, 1013-1042.

Barbosa Povoa, A. P., & Macchietto, S. (1994). Redesign of a multipurpose batch pilot plant with cleaning in place (CIP) integration. Computers & Chemical Engineering, 18, S277-S281.

Bassett, M., Dave, P., Doyle, F., Kudva, G., Pekny, J., Reklaitis, G., Subrahmanyam, S., Miller, D., & Zentner, M. (1996). Perspectives on model based integration of process operations. Computers & Chemical Engineering, 20, 821-844.

Basu, P., Joglekar, G., Rai, S., Suresh, P., & Vernon, J. (2008). Analysis of manufacturing costs in pharmaceutical companies. Journal of Pharmaceutical Innovation, 3, 30-40.

Buchholz, S. (2010). Future manufacturing approaches in the chemical and pharmaceutical industry. Chemical Engineering and Processing: Process Intensification, 49, 993-995.

Calabrese, G. S., & Pissavini, S. (2011). From batch to continuous flow processing in chemicals manufacturing. AIChE Journal, 57, 828-834.

Federsel, H.-J. r. (2009). Chemical Process Research and Development in the 21st Century: Challenges, Strategies, and Solutions from a Pharmaceutical Industry Perspective. Accounts of Chemical Research, 42, 671-680.

Grossmann, I. (2012). Advances in mathematical programming models for enterprise-wide optimization. Computers & Chemical Engineering.

Hynes III, M. D. (2009). Project and capacity management: An application to drug development. Computers & Chemical Engineering, 33, 1994-1998.

Kessel, M. (2011). The problems with today's pharmaceutical business [mdash] an outsider's view. Nature biotechnology, 29, 27-33.

Maravelias, C. T. (2012). General framework and modeling approach classification for chemical production scheduling. AIChE Journal.

Mendez, C. A., Cerda, J., Grossmann, I. E., Harjunkoski, I., & Fahl, M. (2006). State-of-the-art review of optimization methods for short-term scheduling of batch processes. Computers & Chemical Engineering, 30, 913-946.

Moniz, S., Barbosa-Póvoa, A. P., & Pinho de Sousa, J. (2012). Scheduling with equipment redesign in multipurpose batch plants. Foundations of Computer-Aided Process Operations - FOCAPO 2012.

Moniz, S., Barbosa-Póvoa, A. P., & Pinho de Sousa, J. (2013). Simultaneous regular and non-regular production scheduling of multipurpose batch plants: a real chemical-pharmaceutical case study. Under review in Computers & Chemical Engineering.

Moniz, S., Barbosa-Póvoa, A. P., & Pinho de Sousa, J. (2013 submitted). A solution methodology for scheduling problems in batch plants. under review in Industrial & engineering chemistry research.

Pinto, J. M., & Grossmann, I. E. (1998). Assignment and sequencing models for thescheduling of process systems. Annals of Operations Research, 81, 433-466.

Reklaitis, G. V. (1992). Overview of scheduling and planning of batch process operations. Proc. NATO Advanced Study Institute on Batch Processing System Engineering, Antalya, Turkey.

Roberge, D. M., Ducry, L., Bieler, N., Cretton, P., & Zimmermann, B. (2005). Microreactor technology: a revolution for the fine chemical and pharmaceutical industries? Chemical engineering & technology, 28, 318-323.
Roberge, D. M., Zimmermann, B., Rainone, F., Gottsponer, M., Eyholzer, M., & Kockmann, N. (2008). Microreactor technology and continuous processes in the fine chemical and pharmaceutical industry: Is the revolution underway? Organic Process Research & Development, 12, 905-910.
Shah, N. (2004). Pharmaceutical supply chains: key issues and strategies for optimisation. Computers & Chemical Engineering, 28, 929-941.
Shah, N., Pantelides, C., & Sargent, R. (1993). Optimal periodic scheduling of multipurpose batch plants. Annals of Operations Research, 42, 193-228.
Sundaramoorthy, A., & Maravelias, C. T. (2011). A general framework for process scheduling. AIChE Journal, 57, 695-710.
Suresh, P., & Basu, P. K. (2008). Improving pharmaceutical product development and manufacturing: impact on cost of drug development and cost of goods sold of pharmaceuticals. Journal of Pharmaceutical Innovation, 3, 175-187.

3 Paper 1: Scheduling With Equipment Redesign In Multipurpose Batch Plants

Samuel Moniz,[a,c] Ana Paula Barbosa-Póvoa,[a] Jorge Pinho de Sousa[b,c]

[a]*Centro de Estudos de Gestão, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal*

[b]*INESC TEC, 4200-465, Porto, Portugal*

[c]*Faculdade de Engenharia da Universidade do Porto, 4200-465, Porto, Portugal*

This chapter was published in the proceedings of the conference FOCAPO 2012.

Abstract

The objective of this paper is to present a new formulation for the optimal scheduling of multipurpose batch plants where equipment redesign is considered simultaneously with the scheduling decisions. The equipment redesign is characterized by the implementation of modifications in the existent processing units so as to change their suitability to perform certain tasks, while regarding tasks' characteristics inside a given scheduling horizon. This approach may be advantageous in cases where no schedule solutions are found with the existent equipments and where, with minor technology modifications on the processing units, feasible schedules can be obtained. Each of these changes has a cost and requires a certain time to be implemented. In order to model such problem a simple Mixed Integer Linear Programing formulation (MILP) is proposed having as basis the unified Resource-Task Network (RTN) representation presented by Pantelides (1994). An example motivated by a chemical-pharmaceutical industry is used to demonstrate the applicability of the proposed formulation.

Keywords: Multipurpose batch plants, simultaneous scheduling and design, equipment redesign

3.1 Introduction

The chemical-pharmaceutical industry has been facing an increasing demand for the production of a high variety of low volume products at a minimum cost. Such pressure leads to the need of production systems that run efficiently both in terms of cost and time. Consequently, production flexibility is required so as to accommodate the customers' orders within acceptable response times and costs – usually imposed by the market.

To compete in such environment, the chemical industry has been using multipurpose batch plants that are characterized by having a set of resources (processing units, raw materials, utilities, manpower, etc.) that can be shared, so as to produce several products. These plants are especially attractive in situations where product demands and formulations change rapidly, since they can be easily adapted to the production specificities of each product. Moreover, changes in a plant such as the addition of new processing units or connections and the removal of old inefficient units are decisions that can also be considered. In this context, planning and scheduling become important functions of the production system enabling a flexibility increase of the multipurpose batch plants while minimizing costs.

This problem has been addressed in the literature as the design and retrofit of multipurpose batch plants. For the most recent review on these issues see Barbosa-Póvoa (2007). The design of batch plants from scratch is referred as a grassroot problem while the redesign of an existing plant is denoted as a retrofit problem. Two additional concepts have been used to categorize these research problems: "*basic design*" and "*extended design*". As stated by Barbosa-Póvoa (2007) the former refers to the simple choice of equipments and associated scheduling, while the latter goes further and addresses scheduling and detailed design where not only the choice of the equipment is considered but also topology and operational aspects are explored. A number of papers have been published on these topics and the proposed models cover a large number of problem features such as: the selection of the processing units and their sizes; addition of storage vessels; storage policies; design of equipment units' connections; operating mode – cyclic and non-cyclic; campaign structure; and 2D and 3D layout design.

Furthermore, when looking into the batch scheduling problem as a standalone problem, the aim is to operate a set of resources so as to produce a set of products within a defined scheduling period. For a detailed review on this topic the work of Mendez et al.

(2006) should be analyzed. Batch scheduling problems need to deal with a great variety of aspects that are intrinsically linked to the processes and plant structures. Some of the most important of these aspects are: multiproduct and multipurpose batch topologies; equipment connectivity; inventory storage policies; material transfer; batch size and batch processing time; and changeovers. When modeling such problems one of the most important issues is the time representation, which can be discrete or continuous. Discrete formulations have been shown to be a good approach for those scheduling problems that can be represented with a reasonable, not too large, number of time intervals (Castro et al., 2003). Continuous formulations explicitly represent the timing decisions as a set of continuous variables, as a way to define the exact time at which the events occur. Typically, this results in the reduction of the number of variables of the model. Despite the added flexibility, continuous formulations tend to increase the models complexity by means of the use of big-M constraints.

As mentioned before most of the work performed on the scheduling problem of multipurpose batch plants mainly addresses the optimal utilization of a set of existent resources so as to produce what the customers need. On the other hand, the design and retrofit of multipurpose batch plants looks into the need of designing a plant from scratch or redesigning the existing plant, by adding new units or connections. Nevertheless, an intermediate problem, somewhere between the design and the scheduling problem, is often faced by multipurpose process companies when trying to produce a new set of products, see Figure 3.1. This problem is related to the need of performing changes in the existing processing units – equipment redesign – so as to improve the existent equipment suitability, thus providing more flexibility to the plant. The timing of the equipment redesign decisions is similar to the scheduling decisions since their scope is also of short-term. Furthermore, the retrofit and grassroots design take time to be implemented in the shop-floor and may require large investments, hence these decisions must be considered in the long-term planning. The equipment redesign assumes more relevance in industries that perform process development, since the production recipes evolve with it and for that reason it may be necessary to modify the processing units.

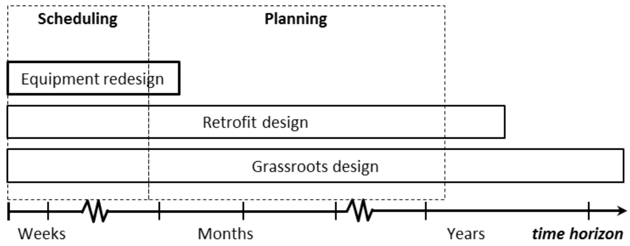


Figure 3.1 - Impact of scheduling, planning and design decisions over the time horizon.

As an example, we have the addition or removal of cleaning-in-place (CIP) systems as well as the addition or removal of temperature or sampling systems. Such operations allow for changes in the equipment’s suitability so as to perform new process recipe tasks. Doing this, new design and scheduling alternatives are then generated at lower cost and with smaller time consumption.

This problem is addressed in the current paper and has emerged from a real problem that is been addressed by the authors in a chemical-pharmaceutical industry. Unlike the previous research on this topic, that has been addressing the plant design as grassroot or retrofit problems at the global plant level, we consider that performing specific changes in the processing units can be an alternative to tackle scheduling and design problems simultaneously. A Mixed Integer Linear Programming (MILP) model is proposed based on the Resource-Task Network (RTN) representation presented by Pantelides (1994).

The remaining of the paper is structured as follows. We first present the problem definition as well as the modeling framework that is being used. Two ways of modeling the equipment redesign problem are then characterized. One uses the original RTN formulation and the other is an extended RTN formulation. We present the computational results of a scheduling problem motivated by the chemical-pharmaceutical industry under study, where equipment “redesign” is a regular approach when performing the production schedule. We finish the paper with the conclusions and some future work is also suggested.

3.2 Problem Definition

As referred above the generic scheduling problem assumes that, when performing scheduling, there must be a perfect match between the tasks requirements and the existent processing units' characteristics. Clearly, this is not easy to do due to the large number of processing units existing in the plant and due to the various recipes requirements. Finding a schedule solution without relaxing any of these inputs is often difficult to accomplish, mainly when the plant operates close to the maximum capacity and when new products are frequently being introduced. In these cases, to get feasible schedules usually requires re-negotiating new order due dates with the customers. Nevertheless, new alternatives for the schedules can also be generated with some equipment modifications involving little costs and time.

The use of multipurpose reactors is indeed advantageous in these situations since such units are very flexible and can often perform several tasks. Additionally, their operating range can be increased by doing small equipment modifications. The same reasoning can be applied to all processing units whose suitability to execute tasks can be changed quickly. The redesign problem takes into account the setup-time to perform the equipment modifications and, at the same time, the resources that are needed to do the modification. This approach transforms the processing units into more generic units capable of executing more tasks. From the point of view of the operations this adds flexibility, since more scheduling alternatives can be explored. Such scheduling with equipment redesign is modeled in the present work and can be described as follows:

Given:

- the RTN representation of the process (tasks and resources);
- the number of processing units available, and their maximum and minimum capacity;
- the scheduling granularity and time horizon;
- the production requirements during the time horizon;
- the auxiliary equipments that can change the suitability of the processing units;
- the cost and setup-time to add and remove auxiliary equipments;

Determine:

- a process schedule such that the processing units suitability change during the time horizon;

- an equipment modification plan to respond to the above schedule, taking into account the setup times for adding and removing the auxiliary equipments and their limited availability;

Minimize:

- the processing units modification costs plus the operational costs, while respecting the delivery due dates.

3.3 Problem Modeling

The problem considered here is modeled with a discrete time formulation based on the Resource-Task Network representation proposed by Pantelides (1994). The scheduling of a set of products is performed in a set of existing equipments allowing for modifications in some resources. The modifications are obtained simultaneously with the definition of the production schedule, within a pre-defined time horizon.

3.3.1 Resource Task Network Discrete Formulation

The Resource-Task Network representation proposed by Pantelides (1994) involves two types of entities, *tasks* and *resources*. A *task* is an abstract operation that consumes and/or produces a specific set of resources (material, equipment items, utilities, etc.). For the purposes of the discrete time formulation presented in this paper, the time discretization is made fine enough so that all tasks can be considered to start and end at a time interval boundary. Each task has a fixed duration τ_k and the execution of task k starting at time t is characterised by its "extent" - a pair of variables (N_{kt},ξ_{kt}). N_{kt} is the number of instances (either 0 or 1) of task k starting at time t while, ξ_{kt} is the total amount of material that is processed by all these instances. Resources are produced and consumed at discrete times, during the execution of the task. The amount of resource r produced or consumed by a task k at different times over its duration τ_k can be obtained from the values of the "extent" variables. Changes to the resource utilisation can occur only at interval boundaries. The amount of unused ("excess") resource r, held over time interval t, is denoted by R_{rt}.

As presented by Pantelides (1994) the RTN discrete scheduling problem can be described by the following three types of constraints:

$$R_{rt} = R_{r,t-1} + \sum_{k \in K_r} \sum_{\theta=0}^{\tau_k} \left(\mu_{kr\theta} N_{k,t-\theta} + \upsilon_{kr\theta} \xi_{k,t-\theta} \right) + \Pi_{rt} \qquad \forall r \in R, t \in H \tag{3.1}$$

$$0 \le R_{rt} \le R_{rt}^{\max} \qquad \forall r \in R, t \in H \tag{3.2}$$

$$V_{kr}^{\min} N_{kt} \le \xi_{kt} \le V_{kr}^{\max} N_{kt} \qquad \forall r \in E, k \in K_r, t \in H \tag{3.3}$$

Constraints (3.1) express resource balancing through the variables R_{rt}, that state the availability of resource r at time t. The amount of resource r consumed and produced at each time is expressed by the integer and continuous part of constraints ($\mu_{kr\Theta}N_{k,t-\Theta}+\nu_{kr\Theta}\xi_{k,t-\Theta}$). $N_{k,t-\Theta}$ is a binary variable that takes the value 1 if task k starts at time t, and $\xi_{k,t-\Theta}$ indicates the amount of material being produced at each time period, *i.e.*, the batch size. The parameters $\mu_{kr\Theta}$ and $\nu_{kr\Theta}$ represent the fixed and variable resource consumption/production, respectively. Constraints (3.2) limit the availability the resources to the maximum value $R_{rt}^{\max}$ during the time horizon. And constraints (3.3) set the batch sizes within the limits of the resource capacity $V_{kr}^{\min}$ and $V_{kr}^{\max}$, where E is the subset of R for the processing units, and K_r is the set of tasks that use resource r.

3.3.2 Equipment Redesign Problem Using the RTN

Applying the existing formulation to the equipment redesign problem requires the explicit representation of all possible modification alternatives. Hence, we need to create new tasks to explicitly take into account all steps required to modify the processing units, *i.e.* to model the addition and removal of auxiliary equipments. This approach will make the network of processing tasks very complex and more difficult to tackle.

Figure 3.2 shows how the RTN formulation can deal with the equipment redesign problem. To consider the setup time for adding and removing the auxiliary equipment CIP on Reactor1, we need to create two additional tasks (Add_CIP and Remove_CIP), and one extra resource (Reactor1_CIP). This allows us to model the availability of Reactor1 after the modification, *i.e.*, having Reactor1 with a CIP system installed.

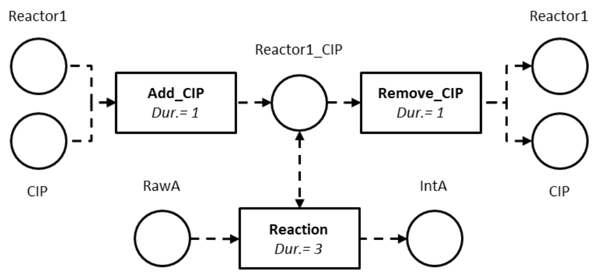


Figure 3.2 - RTN of the equipment redesign problem (reversible modification).

If the modification is irreversible there is no removing task; if the modification is reversible it is necessary to create two tasks: one to add the auxiliary equipment to the processing unit, representing the equipment modification, and another task to remove the previously installed auxiliary equipment, providing the processing unit with its initial suitability. The network of processing tasks requires the explicit representation of all possible combinations of auxiliary equipments (e.g. CIP, sampling devices and temperature systems) and processing units (e.g. reactors, filters, dryers). In the case of the reversible modifications, two additional tasks and one extra resource will be added to the model for each equipment modification needed. For these reasons, the model complexity for representing the problem using the RTN formulation rises.

3.3.3 Equipment Redesign Problem Using an Extended RTN Formulation

An alternative approach to tackle this problem is to create two additional sets of binary variables to control when the processing unit needs to be modified in order to be suitable for the task execution, see constraints (3.4).

$$\begin{aligned} R_{rt} &= R_{r,t-1} + \sum_{k \in K_r} \sum_{\theta=0}^{\tau_k} \left(\mu_{kr\theta} N_{k,t-\theta} + \upsilon_{kr\theta} \xi_{k,t-\theta} \right) + \\ &\sum_{k' \in K_r'} \sum_{u=0}^{s_{k'}} \left(\lambda_{k'ru} M_{k',t-u} + \gamma_{k'ru} \overline{M}_{k',t-u} \right) + \Pi_{rt} \qquad \forall r \in R, t \in H \end{aligned} \tag{3.4}$$

To express the redesign of the processing units, we will use the binary variables M_{kt} and $\overline{M}_{kt}$ that will be equal to 1 if a modification (addition or removal respectively) occurs by means of the task *k* at the time interval *t*. The parameter λ_{kru} denotes the resources *r* that will be consumed (*e.g.*, CIP and Reactor1) by an equipment modification required by a task *k* during the interval *u*, once the modification has started. The parameter γ_{kru} denotes the reverse operation. It consumes the modified resource (*e.g.*, Reactor1) and releases back the resources (*e.g.*, CIP and Reactor1). The setup-time required for each modification is given by the parameter s_k. Constraints (3.1) are modified and a third term is added to reflect this behavior. The $(\lambda_{kru} M_{k,t-u})$ expression enforces the modifications to be done by each task k, while the $(\gamma_{kru} \overline{M}_{k,t-u})$ part denotes the removal of the auxiliary equipment from the processing units.

The entire formulation also guarantees that the auxiliary equipment cannot be removed during the task execution and that the setup-times s_k for modifying the processing units are respected. K'_r is a subset of K_r that denotes the tasks that require redesign through the auxiliary resources r. More specifically, for the example given in Figure 3.2, we get the $\lambda_{Reaction,Reactor1,0} = \lambda_{Reaction,CIP,0} = -1$ and $\lambda_{Reaction,Reactor1,1} = 1$ and $\gamma_{Reaction,Reactor1,0} = -1$ and $\gamma_{Reaction,Reactor1,1} = \gamma_{Reaction,CIP,1} = 1$, see Figure 3.3.

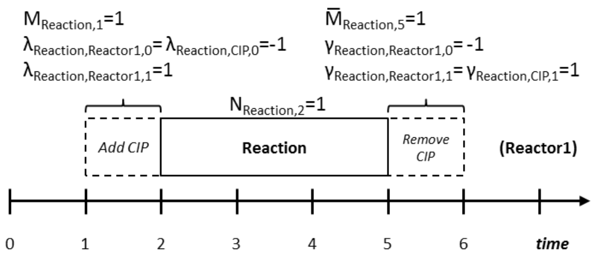


Figure 3.3 - Equipment redesign modeling with the alternative formulation.

An additional constraint type is also needed for the correct assignment of the M_{kt} and $\overline{M}_{kt}$ binary variables. Since the equipment modification needs to be done before the task starts, constraints (3.5) guarantee that the auxiliary equipment has been previously installed. A is the subset of R which has auxiliary equipments needed to modify the processing units, and K_{kr} is the set of tasks that share the auxiliary equipment r.

$$N_{k,t} \leq \sum_{\theta=0}^{t} \sum_{k' \in K_{kr}} \left(M_{k',\theta} - \overline{M}_{k',\theta} \right) \quad \forall r \in A, k \in K_r, t \in H \tag{3.5}$$

When the binary variables N_{kt} are equal to 1, the right hand side of the constraints needs also to be 1, therefore having at that time instant a sum (involving the M_{kt} and $\overline{M}_{kt}$ variables) equal to 1. In practice, this means that the auxiliary equipment needs to be previously consumed by that task, or by other task that was executed in the past and that required the same auxiliary equipment in the same processing unit.

With this formulation, there is no need to explicitly write the modification tasks. Instead two sets of additional binary variables are added to the model to express the addition and removal of auxiliary equipments to the processing units. The resources are still treated uniformly as they are in the original RTN formulation.

Finally, for both formulations the objective function considered in this work is the minimization of the processing units modification costs C_k and, $\overline{C}_k$ as well as the operational costs O_k, see equation (3.6).

$$\min\left[\sum_{k\in K}\sum_{t\in H}\left(O_k N_{k,t}\right)+\sum_{k\in K^{'}}\sum_{t\in H}\left(C_k M_{kt}+\overline{C}_k \overline{M}_{kt}\right)\right] \tag{3.6}$$

3.4 Case Study

A real world problem from a chemical-pharmaceutical industry is solved using both presented formulations. The company performs the development and production of complex and fine chemicals to the pharmaceutical industry and biotechs. Its core business is the development and manufacture of new active pharmaceutical ingredients (APIs). In this business, the chemical industry is continuously challenged to respond within short time windows. On the one hand, the company needs to manage small batches of under development products and, on the other hand, needs to produce large batches of products in commercialization. Thus, operations flexibility is required to respond to this heterogeneous demand. This adds extra complexity to operations management especially to the planning and scheduling functions.

The product object of our analysis goes through a sequence of tasks such as reaction, precipitation, crystallization, filtration, suspension, drying, quality control and packaging, which can be performed by the following resources: four reactors, one vessel, one filter, one dryer and a packaging room. The typical production time is around ten days. For illustration purposes, we will focus here on the multipurpose reactors since these are the most difficult resources to schedule, thus imposing the schedule of the remaining resources. Devices such as CIP and temperature systems (TS) are considered auxiliary equipments that can be used for the reactors redesign. The reaction, precipitation, crystallization and suspension tasks can either be executed in reactors that do not require modifications but have small capacity, or can be executed in reactors with higher capacity but need to be modified at a certain cost. The product must be delivered at a date and quantity agreed with the customer. The objective is to get the optimal schedule for this product, minimizing the global operation and modification costs, while respecting the product delivery date.

3.4.1 Case Study Results

The scheduling problem was solved for a time horizon of ten days. The time was discretized to one shift of eight hours, which resulted in a scheduling horizon of 30 time intervals (three shifts per day). We have considered an operational cost for each task depending on the processing unit that is used. Tasks that take place in low capacity reactors (capacity of 4,000 liters) have an operational cost of 70 m.u. (monetary units) and tasks that are performed in high capacity reactors (capacity of 10,000 liters) have a cost of 100 m.u..

In the course of the recipe production the tasks' characteristics may change requiring the processing units redesign. For instance, precise temperature control is needed on Mixing and Precipitation tasks at Reactor1 and Reactor2, and a CIP system must be available in Reactor2 and Reactor3 when performing Reaction and Stirring tasks, respectively. The costs to modify a reactor with a CIP and TS are, respectively, 3 m.u. and 5 m.u.. The setup-time to modify the reactors with a CIP is 8 hours, while for a TS is 16 hours. The time required to remove those systems from the reactors in order to restore their original suitability is equal to 8 hours for both auxiliary equipments. One final product delivery of 2 tons is scheduled for the entire schedule horizon. The optimal schedule obtained for our example is depicted in Figure 3.4. This optimal solution has a value of 2074 m.u. Although this test instance is relatively simple, it allows us to understand the tradeoffs existing in the equipment redesign problem, between equipment's suitability and the setup-time and costs to perform the equipments modifications. As can be seen in Figure 3.4, to respect the delivery date, equipment redesign tasks must take place. To perform the Reaction task in Reactor2 it is necessary to add a CIP, and to do the Precipitation task in this same reactor it is necessary to add a TS. These tasks can be seen at the time interval 0 and 5 of the schedule, respectively. The same reasoning applies to the Mixing task at Reactor1 and to the Stirring task at Reactor3. But note that no auxiliary equipments were defined for the Crystallization task at Reactor2 and for the Cooling task at Reactor3, that nevertheless were modified previously. In the end of this schedule Reactor2 had a TS installed, while Reactor3 had a CIP mounted. The MILP model using Pantelides formulation resulted in 1178 binary variables, 2202 continuous variables and 5085 constraints. Optimality could be proved in 3.15 seconds. The extended formulation has 775 binary variables, 1396 continuous variables, 2853 constraints and reached the optimal solution in 1.78 seconds.

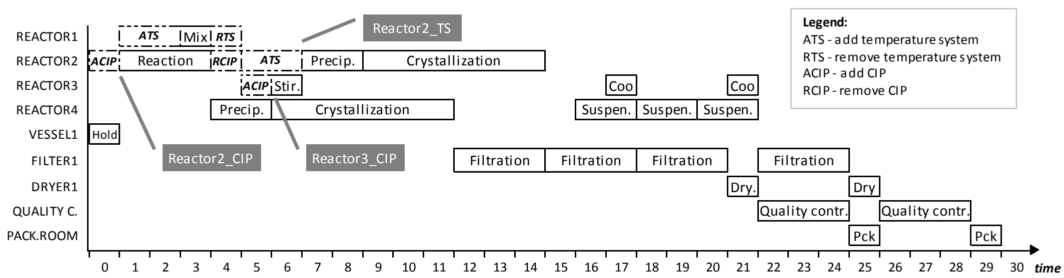


Figure 3.4 - Optimal production schedule with the equipment redesign plan.

The model was implemented using ILOG/CPLEX version 12.2 on an Intel Core i7 at 2.67GHz with 4 GB of RAM. The extended formulation has less binary and continuous variables and a smaller number of constraints.

When analyzing these results some disadvantages can be pointed to the original RTN formulation when using it in the redesign problem. It requires the representation of all modification tasks, which results in a complex network of processing tasks. One needs to create additional resources to manage the modified equipments, such as for instance: Reactor2_CIP and Reactor2_TS; these are two additional resources that define Reactor2 modified with a CIP and a TS, respectively. At the same time, since we are assuming the redesign process increases the processing units' suitability such that more tasks can be performed, we must represent all new production alternatives. For instance, the Crystallization task does not require any change on Reactor2, nevertheless if this reactor is modified with a CIP or TS, becoming Reactor2_CIP, Reactor2_TS or Reactor2_CIPTS, we need to create several additional tasks to allow for the possibility of the task being executed in one of these resources. This kind of tasks needs to be created for all resources that can be modified, thus increasing the model size. These drawbacks are overcome in the proposed formulation by replacing the redesign tasks by M_{kt} and $\overline{M}_{kt}$ binary variables. The resulting model is smaller and it is easier to write since it does not require the representation of additional tasks. The redesign tasks are simply modeled by the M_{kt} and $\overline{M}_{kt}$ variables. For that reason, the resulting MILP has less binary and continuous variables. Nevertheless, the use of the M_{kt} and $\overline{M}_{kt}$ variables limits the equipment modification to one auxiliary equipment per task. The possibility of doing more than one modification per task would clearly be an interesting extension of our model.

3.5 Conclusions

This paper has addressed a new type of problem that is being faced by the chemical-pharmaceutical industry using multipurpose batch plants, and performing simultaneous design and scheduling within a short period of time.

The equipment redesign problem concerns the need to perform changes in the processing units such that their suitability is increased and therefore the units are capable to perform additional tasks. The redesign tasks can be seen as an additional way to increase flexibility of these plants. The redesign problem was formulated using the RTN formulation introduced by Pantelides and an extension to this formulation was also proposed in this work. While the RTN formulation requires the explicit representation of all production alternatives, taking into account the different states of the modified resources, the extension here developed deals with the equipment redesign decisions through two extra groups of binary variables. Preliminary computational results show that the proposed formulation has better performance. The formulation applicability was tested in an industrial example and the achieved results are promising but improvements should be further explored. Namely, it would be interesting to extend the formulation to deal with multiple modifications per task. Also more comprehensive tests need to be performed to further compare the two analyzed formulations.

References

Barbosa-Póvoa, A. P. (2007). A critical review on the design and retrofit of batch plants. Computers & Chemical Engineering, 31, 833-855.

Castro, P. M., Barbosa-Póvoa, A. P., & Matos, H. A. (2003). Optimal periodic scheduling of batch plants using RTN-based discrete and continuous-time formulations: A case study approach. Industrial & engineering chemistry research, 42, 3346-3360.

Mendez, C. A., Cerda, J., Grossmann, I. E., Harjunkoski, I., & Fahl, M. (2006). State-of-the-art review of optimization methods for short-term scheduling of batch processes. Computers & Chemical Engineering, 30, 913-946.

Pantelides, C. C. (1994). Unified frameworks for optimal process planning and scheduling. In (pp. 253-274): Cache Publications New York.

4 Paper 2: New General Discrete-Time Scheduling Model for Multipurpose Batch Plants

Samuel Moniz,[a,c] Ana Paula Barbosa-Póvoa,[a] Jorge Pinho de Sousa[b,c]

[a]*Centro de Estudos de Gestão, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal*

[b]*INESC TEC, 4200-465, Porto, Portugal*

[c]*Faculdade de Engenharia da Universidade do Porto, 4200-465, Porto, Portugal*

This chapter was published in the journal Industrial & Engineering Chemistry Research.

Abstract

This work deals with the optimal short-term scheduling of general multipurpose batch plants, considering multiple operational characteristics such as sequence-dependent changeovers, temporary storage in the processing units, lots blending, and material flows traceability. A novel Mixed Integer Linear Programming (MILP) discrete-time formulation based on the State-Task Network (STN) is proposed, with new types of constraints for modeling changeovers and storage. We also propose some model extensions for addressing changeovers start; non-preemptive lots; lots start and sizes; alternative task-unit and task-unit-layout assignments. Computational tests have shown that the proposed model is more effective than a similar model based on the Resource-Task Network (RTN).

Keywords: Multipurpose batch plants, scheduling, MILP models, lots modeling, materials traceability, Resource-Task Network.

4.1 Introduction

In the past decades many optimization approaches have been developed to address supply chain, planning, and scheduling problems. These developments are being motivated by the need that industries have to reduce costs, increase revenues, and in general, to operate more efficiently. Consequently, the existing gap between theoretical optimization models and real world scheduling problems is gradually decreasing. This can be somehow justified by the increasing number of works published in the recent years that address practical optimization problems. According to Grossmann (2012), process industries are actively looking for optimization approaches that can be integrated in key decision-making processes so as to minimize costs and maximize income, while increasing the system responsiveness. In the particular case of the pharmaceutical industry, Varma et al. (2007) argue on the importance of developing models that integrate decision-making processes related to R&D, manufacturing, supply chain, and marketing. An extensive review on the modeling approaches for scheduling problems that tackle these issues is available in the paper written by Mendez et al. (2006), where the characteristics, advantages, and disadvantages of the models are deeply addressed.

Although a significant progress has been observed in this field, new planning and scheduling models are still needed to tackle existing complexities that remain unsolved and to address new challenges that are becoming more relevant. In this paper, we propose a short-term scheduling model for multipurpose batch plants that addresses two critical modeling features of the discrete-time models: the sequence-dependent changeovers and the temporary storage in the processing units. We also discuss lots blending and traceability requirements in the production schedules. Particular emphasis is given to the performance of the proposed model. The consideration of such aspects was motivated by the resolution of a real case study within the chemical-pharmaceutical industry that led to the design of an illustrative problem instance, used to assess the developed models.

The rest of the paper is structured as follows. In section 4.2, we describe an example to illustrate the impact of the definition of lots in the production schedule, and in section 4.3, a literature review is presented. The problem statement is introduced in section 4.4, and it is followed by the mathematical formulations in section 4.5. Then in section 4.6, we propose several models extensions, and in section 4.7, we compare the models performance. Finally, section 4.8 provides some concluding remarks.

4.2 Illustrative Example

This example is motivated by a case study occurring in a real chemical-pharmaceutical industry where it is critical to consider some production features such as sequence-dependent changeovers, temporary storage in the processing units, lots blending, and materials traceability.

Consider the determination of a production schedule for three products: PA, PB, and PC. Task sequences and respective alternative units are depicted in Figure 4.1. Products PA and PB are produced from raw materials, while product PC is produced from PA and PB. The objective is to maximize the overall profit by determining a schedule that keeps record of the production lots and involves sequence-dependent changeovers between products and lots.

A distinction is made between *lots* and *task - batches*. The former have to do with the amount of stable intermediary or final product produced through a known set of tasks, processing units and materials. The latter are related to the amount of material produced by each task that is limited by the capacity of the processing unit and is part of the production of a lot.

In this way, lots traceability must be ensured for all products considered in the production schedule. We must be able to trace the proportions/quantities of the lots of PA and PB used to produce each lot of PC. This means that lots blending may occur and that the scheduling model must consider the amount of each lot used to produce subsequent lots. Raw materials and intermediaries must be also associated to lots. Generally, the scheduling model must do the record (*i.e.*, allow for traceability) of the task-batching (materials splitting and mixing) and of the lots blending process.

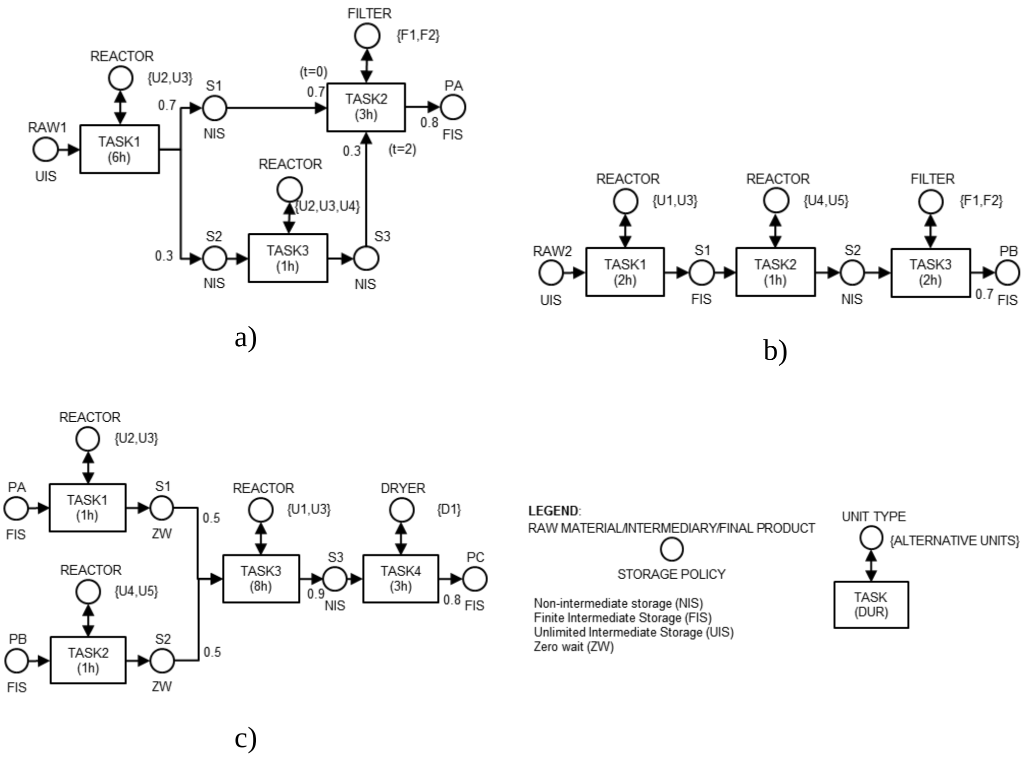


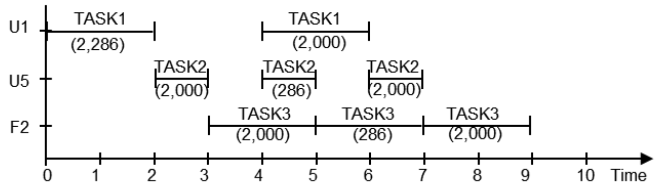
Figure 4.1 – Illustrative example.

In order to illustrate the impact of production lots on scheduling, we consider a small instance with product PB for a scheduling horizon of 10 hours. We want to determine a production schedule in which the task-unit assignment accounts for a given lot size and that the lots traceability is ensured. For that, we define a fixed demand equal to 3,000 kg that is produced assuming two scenarios. The first is a base scenario where no lots are defined, while in the second scenario we assume two lots of 1,500 kg.

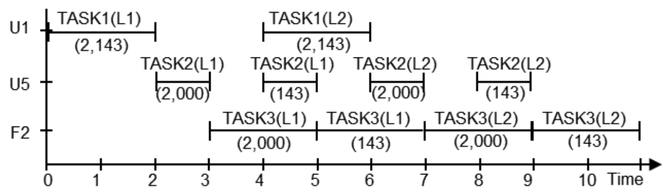
In Figure 4.2 a), we show a schedule for the base scenario, and as it can be seen, the tasks batch size is as large as possible, so as to minimize the number of tasks and therefore the production costs. The amount of material produced by two tasks TASK1 is split by three tasks TASK2 and three tasks TASK3. Since lots were not explicitly modeled, it is not possible to make a task-lot assignment; thus, the schedule of Figure 4.2 a) does not account for lot traceability.

On the contrary, the schedule depicted in Figure 4.2 b) results in the same amount of final product, but considers lots traceability. The difference is in the number and respective batch sizes of the tasks. To consider lots traceability the schedule must have unique associations between tasks and lots. In our example, it can be seen that the first task TASK1 and the two first tasks TASK2 and TASK 3 are associated to lot L1, while the other tasks are associated to lot L2. In this way, raw materials, intermediaries and final products are distinctively associated to each lot. The impact of lots in scheduling would be higher if sequence-dependent changeovers were considered.

In dynamic production environments lots are bound by minimum and maximum sizes, and the exact size of each lot is just determined when performing scheduling. This is done to ensure that the processing units are used as efficiently as possible.



a)



b)

Figure 4.2 – a) Schedule assuming a demand of 3,000 kg with no defined lots; b) Schedule with two lots of 1,500 kg.

4.3 Background

Scheduling of process plants has received considerable attention in the literature, with some relevant reviews on the topic (Grossmann, 2002; Mendez et al., 2006; Barbosa-Povoa, 2007; Li & Ierapetritou, 2008; Maravelias & Sung, 2009; Verderame et al., 2010).

Scheduling problems can be classified in terms of the network of processing tasks (Mendez et al., 2006). The allowed material flow and unit specific constraints strongly determine the modeling approach and, consequently, the model performance and its complexity. In general, we may have sequential or network processes.

In sequential processes the batch entity is preserved by ensuring that the output of a batch is consumed by a single task and the input of a batch is produced by a single batch. Within the sequential processes, single and multiple stage topologies can be defined. The former consists in production systems with just one stage and may have parallel units, and the latter involves production systems with more than one stage that may also have parallel units. Sequential processes can either use precedence-based or time-grid formulations. Precedence-based models have been proposed by several authors (Méndez et al., 2001; Méndez & Cerdá, 2003; Castro, Erdirik-Dogan, et al., 2008; Sundaramoorthy & Maravelias, 2008) and time-grid models for sequential processes rely on time-slots (Pinto & Grossmann, 1995; Liu & Karimi, 2007, 2008).

On the contrary, network processes have an arbitrary topology and are usually more complex than sequential processes, since they deal with batch mixing and splitting and cyclic material flows. For these reasons, models for network topologies require resource balance constraints and are time-grid based, either discrete-time or continuous-time. By definition models used for network processes can also be applied to sequential processes, since they can model all types of process configurations. Continuous-time formulations may rely on unit specific events (Ierapetritou & Floudas, 1998; Janak et al., 2004; Shaik & Floudas, 2007; Vooradi & Shaik, 2012) or on global events (Schilling & Pantelides, 1996; Maravelias & Grossmann, 2003; Castro et al., 2004; Sundaramoorthy & Karimi, 2005; Castro et al., 2009). The major advantage of the continuous-time formulations is that tasks may occur anywhere in the scheduling horizon and thus these models are considered more accurate. However, in terms of mathematical programming, continuous-time models generally result in large integrality gaps that tend to deteriorate computational times.

Discrete-time formulations assume that the scheduling horizon has been divided into a finite number of time intervals of fixed and equal duration. Tasks are allowed to take place just in the boundaries of the time intervals, which makes it easier to model inventory and units availability constraints. These models deal easily with material balances and inventory costs, and multiple delivery dates and result into compact formulations. On the other hand, they present some difficulties when modeling variable processing times and sequence-dependent changeovers. Moreover, we need to be aware of the tradeoffs between accuracy of the scheduling solutions, the time discretization, and the scheduling horizon, since computational performance strongly depends on the number of time intervals considered. Both the State-Task Network (STN) representation suggested by Kondili et al. (1993) and Shah et al. (1993), and the RTN representation introduced by Pantelides (1994) have been widely used for modeling schedule problems. For example, Barbosa-Povoa and Macchietto (1994) developed the Maximal State-Task Network (m-STN) representation that simultaneously considers operational and design characteristics. Pinto et al. (2005) modified RTN to address design and retrofit of batch plants with periodic mode operation. Castro, Novais, et al. (2008) solved an industrial scheduling problem from the chemical-pharmaceutical industry by proposing a periodic RTN formulation. Wassick and Ferrio (2011) proposed some extensions for RTN. Sundaramoorthy and Maravelias (2011b) developed a scheduling framework that addresses the recipes structure in network and sequential subsystems. And more recently, Moniz et al. (2012) proposed a sequential approach for the simultaneous scheduling of regular and non-regular products in multipurpose-batch plants. The integrated approach is based on RTN and is applied to a real scheduling problem from the chemical-pharmaceutical industry. For a comparison of discrete-time and continuous-time models see (Floudas & Lin, 2004; Sundaramoorthy & Maravelias, 2011a).

4.4 Problem Statement

In this paper, we address the short-term scheduling of multipurpose batch plants dealing with products having arbitrary network processes. All product recipes are given in terms of their respective RTNs and may involve sequence-dependent changeovers, materials storage, mixing and splitting operations, and material recycles flows. Product/lots demands are defined for multiple delivery periods and have an earliest and latest delivery date. The characteristics of the processing units, maximum and minimum capacity, operational costs, and the task-unit suitability are assumed to be known. We also assume that the value of the products and the storage costs for all materials (intermediaries and products) are given. The raw materials are the exception, since we consider that they are available as needed. All data is assumed to be deterministic.

The objective is to maximize the economical result of the global operation by determining the task-unit-layout assignment, the tasks sequencing and corresponding batch size, the sequence-dependent changeovers, the temporary storage in the processing units and eventual lots blending needs.

4.5 Mathematical Formulations

4.5.1 Concepts and Notation

In order to compare the effectiveness of the proposed formulation (denoted later in this work by model M2), we present an additional mathematical formulation (model M1) based on the RTN formulation of Pantelides (1994), where scheduling aspects studied by other authors are incorporated in an integrated form. Variations of M1 formulation, in their discrete-time form, have been extensively used by other authors such as Castro et al. (2003), Castro, Novais, et al. (2008), and Wassick and Ferrio (2011).

The key differences between the models are that M1 explicitly models the changeover and storage tasks and does not account for lots blending, while M2 implicitly considers changeovers and storage and accounts for lots blending and traceability features. Additionally, model M1 allows the definition of resource types; thus, processing units with the same characteristics (*e.g.*, minimum and maximum capacity) can be grouped, which leads to a reduction of the number of binary variables, when compared with model M2. Nevertheless, task-unit assignment variables in M1 imply that tasks are performed by single units at each time interval, therefore for handling alternative units they must be considered individually.

Products can be delivered within a given time window, in amounts modeled as "soft constraints" to ensure that feasible schedules are always obtained.

The formulations use the indices, sets, parameters and variables presented below. The exact meaning of each element will be explained later with the formulations.

Indices

d	delivery period
l	lot
k	task
p	product
r	resource (processing unit, intermediary or final product)
t	time interval

Sets

A_k	alternative tasks for task *k*
B	resource *r* (intermediary or final product) in which the lots that can be blended
B_r	lots from resource *r* (intermediary or final product) that can be blended
D_r	delivery periods of resource *r* (final product)
DW_{rld}	delivery window of lot *l* and resource *r* (final product) at delivery period *d*
E	processing units
f_l^r	tasks associated to processing unit *r* and lot *l*
H	scheduling horizon
I	intermediaries
I^{NIS}	intermediaries subject to a non-intermediate storage policy
I_k^{NIS}	intermediaries produced by task *k* and subject to a non-intermediate storage policy
L	lots
L_r	lots associated with resource *r*
L_k	lots associated with task *k*
K_r	tasks that require resource *r* (processing unit, intermediary or final product)
K_r^c	tasks that consume resource *r* (intermediary or final product)
K_r^p	tasks that produce resource *r* (intermediary or final product)
K_r^{sto}	storage tasks associated with intermediary *r*
P	products
R	production resources
S	task *k* that follows task *k'* at adjacent processing units

Parameters

$\alpha_{rl'l''l\theta}$	allocation/release changeover coefficient of resource *r* (processing unit) from lot *l'* to *l''* being at lot *l* and at time θ relative to the start of the changeover task
$\mu_{kr\theta}$	allocation/release coefficient of resource *r* (processing unit) in task *k* at

	time θ relative to the start of the task
τ_k	processing time of task k
$\nu_{kr\theta}$, $\nu^p_{kr\theta}, \nu^c_{kr\theta}$	production/consumption proportion of resource (intermediary or final product) r in task k at time θ relative to the start of task
c^{sto}_r	cost of storage of products and intermediaries r
c^{op}_k	operational costs of task k
c^{slack}_{rld}	missing deliveries cost for material r of lot l and delivery d
$c_{rll'}$	changeover time in processing unit r from lot l to lot l'
$Q^{min}_{rld} Q^{max}_{rld}$	minimum and maximum amount of lot l and product r at delivery d
R^{max}_{rt}	maximum resource availability of resource r (intermediary or final product) at time interval t
$R^{init(m)}_{rl}$	resource r (intermediary or final product) availability of lot l in the beginning of the planning horizon
$R^{init(m)}_{krl}$	resource r (intermediary or final product) availability of lot l at task k in the beginning of the planning horizon
T	length of the scheduling horizon
T^{ed}_{ld}	earliest time interval of lot l at delivery d
T^{dd}_{ld}	latest time interval of lot l at delivery d
v_r	value of product r
$V^{min}_{krl}, V^{max}_{krl}$	minimum and maximum capacity of resource r (processing unit) for task k of lot l

Variables

ξ_{klt}	batch size of task k and lot l at time interval t (continuous)(models M1 and M2)
Π_{rlt}	delivery of resource (final products) r of lot l at time interval t (continuous) (model M1)
Π_{krlt}	delivery of resource (final products) r of lot l at time interval t available from task k (continuous) (model M2)
Π^{slack}_{rld}	missing delivery d of lot l of product r (continuous) (models M1 and M2)
$C_{rll't}$	binary variables that are equal to 1 if a changeover task occur on resource (processing units) r between lots l and l'(model M1)

N_{klt}	binary variables that are equal to 1 if task *k* starts lot *l* at time interval *t* (models M1 and M2)
R_{rl}^{init}	allocation of resource *r* (processing unit) at the beginning of the scheduling horizon (continuous) (model M1)
R_{rlt}	resource availability *r* at lot *l* and at time interval *t* (continuous) (model M1)
R_{rt}	resource availability (processing units) *r* at time interval *t* (continuous) (model M1)
R_{krlt}	resource *r* (intermediaries or final products) availability, produced by task *k* of lot *l* at time interval *t* (continuous) (model M2)
R_{krlt}^{c}	amount of resource *r* (intermediaries or final products) consumed from task *k* of lot *l* at time interval *t* (continuous) (model M2)
R_{krlt}^{p}	amount of resource *r* (intermediaries or final products) produced by task *k* of lot *l* at time interval *t* (continuous) (model M2)
X_{kl}	binary variables that are equal to 1 if task *k* is assigned to lot *l* (models M1 and M2)

4.5.2 RTN Model (M1)

We use a RTN discrete-time formulation as basis for comparison with the model proposed in this paper. Model M1 extends the RTN model of Pantelides (1994) by considering the temporary storage in the processing units constraints defined by Kondili et al. (1993), the changeover variables proposed by Castro, Novais, et al. (2008) and the multiproduct delivery extensions developed by Wassick and Ferrio (2011). Moreover, in section 4.6, we also propose some extensions to address the start of changeovers tasks, non-preemptive lots, lots start and sizes, task-unit-layout assignment, and alternative task-unit assignment.

We assume a scheduling horizon having a length equal to T and divided into time intervals of fixed length. The model considers the following decision variables that are defined for each time interval $t \in H$.

a) The assignment of tasks to processing units decisions is done by the N_{klt} binary variables that are equal to 1 if task *k* starts lot *l* at time interval *t*.
b) The task batch size decisions are done through the ξ_{klt} continuous variables that define the batch size of task *k* and lot *l* at time interval *t*.

c) Changeover tasks are defined by the binary variables $C_{rll't}$ that are equal to 1 if a changeover task occurs on resource (processing unit) *r* between lots l and l' at time interval *t*.

d) Resources availability is given by the $\mathrm{R_{rlt}}$ continuous variables that define the resource availability *r* at lot *l* and at time interval *t*.

e) Deliveries are modeled by the Π_{rlt} continuous variables that define the delivery of resource (final products) *r* of lot *l* at time interval *t*. If the minimum demand is not fulfilled, then the $\Pi_{\mathrm{rld}}^{\mathrm{slack}}$ continuous variables will have a value equal to the amount that was not delivered.

R_{rl}^{init} variables are used to model the initial allocation of processing units to lots. In the cases where changeovers are not required, we use the R_{rt} continuous variables that define the resource availability (processing units) *r* at time interval *t*.

Model M1 considers the processing units with changeovers constraints (4.A1) and the initial assignment of processing units to lots constraints (4.A2), or alternatively, the processing units balance without changeovers constraints (4.A3); materials balance constraints (4.A4); minimum and maximum materials availability constraints (4.A5); minimum and maximum task batch size constraints (4.A6); temporary storage in the processing units constraints (4.A7); demand constraints (4.A8); delivery constraints (4.A9) and (4.A10); tasks started must end within the time horizon constraints (4.A11), and variables domain constraints (4.A12). Model M1 formulation is given in the appendix.

4.5.3 Proposed Model (M2)

Discrete-time models efficiently deal with resources balances, multiple delivery dates, and inventory costs. However, the model size significantly increases and the computational performance is seriously affected when modeling variable processing times, temporary storage in the processing units, and sequence-dependent changeovers.

The storage in the processing units is commonly used in many industrial processes due to the multipurpose characteristics of the units. In these situations, intermediaries can be stored temporarily inside the processing units that have produced them. In practice, this type of storage may be required for a variety of reasons. Some of the possible cases are: a) the capacity of the processing units that follow in the process may be low when compared with the amount of material being stored; b) the lot may need to wait for quality approval; c) scheduling delays may occur, forcing intermediaries to wait temporally in the processing units; or d) maintenance tasks may be required, also imposing scheduling delays.

Changeovers cannot be neglected since they often occupy processing units during long time periods. We may have unit and sequence-dependent changeovers, the latter being usually more significant in terms of time. Sequence-dependent changeovers can be modeled in the original RTN formulation through the creation of changeover tasks, as done in model M1, or if it is not relevant to determine the exact time of the changeover, we can use changeover constraints.

In order to avoid increasing the number of binary variables of the model, as a result of modeling temporary storage and changeovers, we have developed a new discrete-time formulation. The developed model also addresses lots blending and traceability features.

This model explicitly considers the inventory carried out by each production task. Following this approach, we can model the temporary storage through a set of constraints instead of using additional binary variables as done in model M1. Regarding sequence-dependent changeovers, we have followed a similar strategy. Changeover variables are replaced by a set of constraints that inhibit the start of the production tasks for a time period imposed by the changeover time of the tasks sequence.

Figure 4.3 shows the conceptual differences between models M1 and M2 for the resource availability variables. While in M1 all resources are treated uniformly through the continuous variables R_{rlt}, in M2 the continuous variables R_{krlt} define the amount of resource *r* (intermediaries or final products) available at time interval *t* and produced by task *k* of lot *l*.

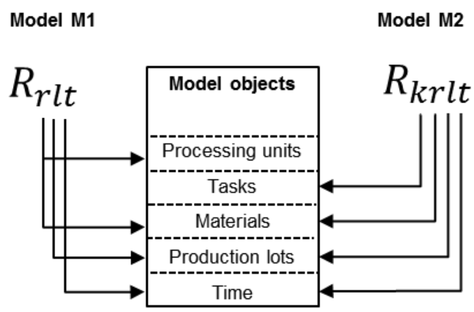


Figure 4.3 – Resource availability variables for models M1 and M2.

The relations between products, lots, tasks, and units sets are illustrated in Figure 4.4. We assume that we have a set of products *P*; in the example we have $\{\text{PA1}, \text{PA2}, \text{PB}\}$, associated with recipes that describe the tasks sequence, the task-unit suitability, the materials needs, and the storage policies. A recipe may involve the production of one or more products. In the example shown in Figure 4.4, products PA1 and PA2 are sub-products of a unique recipe. Each product has at least one lot belonging to set *L*. Production tasks are associated to processing units and belong to set *K* and may execute any lot of the corresponding product. Finally, processing units belong to set *E* and are associated to different tasks, since they operate in a multipurpose way.

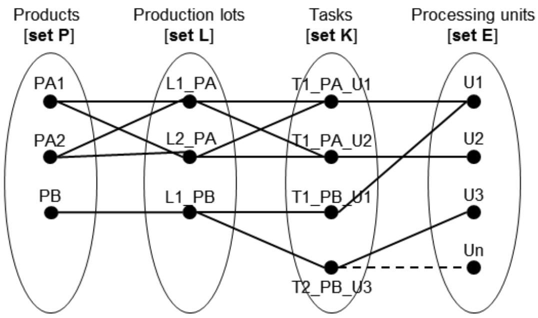


Figure 4.4 – Relation between products, lots, tasks, and units sets.

Model M2 is defined by task-unit assignment/sequencing constraints (4.1); materials produced and consumed, constraints (4.2) and (4.3) respectively; products blending constraints (4.4), materials balance constraints (4.5), minimum and maximum materials availability constraints (4.6); minimum and maximum task batch size constraints (4.7); demand constraints (4.8); delivery constraints (4.9) and (4.10); temporary storage in the processing units without or with changeovers, constraints (4.11) and (4.12) respectively; tasks started must end in the time horizon constraints (4.13); and variables domain constraints (4.14).

Constraints

$$\sum_{k\in K_r}\sum_{l\in L_r}\sum_{t'=t-\tau_k+1}^{t} N_{klt'} \leq 1 \quad \forall r\in E, t\in H \tag{4.1}$$

$$R_{krlt}^{p} = \sum_{\theta=0}^{\tau_k}\left(\nu_{kr\theta}^{p}\xi_{kl,t-\theta}\right) \quad \forall r\in I\cup P, k\in K_r^p, l\in L_r, t\in H \tag{4.2}$$

$$\sum_{k\in K_r^p} R_{krlt}^{c} = \sum_{k\in K_r^c}\sum_{\theta=0}^{\tau_k}\left(\nu_{kr\theta}^{c}\xi_{kl,t-\theta}\right) \quad \forall r\in I\cup P\backslash B, l\in L_r, t\in H \tag{4.3}$$

$$\sum_{k\in K_r^p}\sum_{l\in B_r} R_{krlt}^{c} = \sum_{k\in K_r^c}\sum_{l\in L_k}\sum_{\theta=0}^{\tau_k}\left(\nu_{kr\theta}^{c}\xi_{kl,t-\theta}\right) \quad \forall r\in B, t\in H \tag{4.4}$$

$$R_{krlt} = \left(R_{krl}^{init(m)}|_{t=0}, R_{krl,t-1}|_{t>0}\right) + R_{krlt}^{p} - R_{krlt}^{c} + \Pi_{krlt} \quad \forall r\in I\cup P, k\in K_r^p, l\in L_r, t\in H \tag{4.5}$$

$$0 \leq \sum_{k\in K_r^p}\sum_{l\in L_r} R_{krlt} \leq R_{rt}^{max} \quad \forall r\in I\backslash I^{NIS}\cup P, t\in H \tag{4.6}$$

$$V_{krl}^{min}N_{klt} \leq \xi_{klt} \leq V_{krl}^{max}N_{klt} \quad \forall\, r\in E, k\in K_r, l\in L_k, t\in H \tag{4.7}$$

$$Q_{rld}^{min} - \Pi_{rld}^{slack} \leq \sum_{k\in K_r^p}\sum_{t\in DW_{rld}}\left(-\Pi_{krlt}\right) \leq Q_{rld}^{max} \quad \forall r\in P, l\in L_r, d\in D_r$$
$$DW_{rld} = \left\{t \mid r\in P, l\in L_r, d\in D_r, t\in H : T_{rd}^{dd} \geq t \geq T_{rd}^{ed}\right\} \tag{4.8}$$

$$\Pi_{krlt} = 0 \quad \forall r\in P, l\in L_r, d\in D_r, k\in K_r^p, t\in H\backslash DW_{rld} \tag{4.9}$$

$$\Pi_{krlt} = 0 \quad \forall r\in I, L\in L_r, k\in K_r^p, t\in H \tag{4.10}$$

$$\sum_{k\in K_r}\sum_{l\in L_r} N_{klt} + \sum_{k\in K_r^p}\sum_{r'\in I_k^{NIS}}\sum_{l\in L_r}\left(\frac{R_{kr'lt}}{V_{kr'l}^{max}}\right) \leq 1 \quad \forall r\in E, t\in H \tag{4.11}$$

$$\sum_{k\in f_l^r} N_{klt} + \sum_{k'\in f_{l'}^r} N_{k'l't-\tau_{k'}-\theta+1} + \sum_{k''\in f_{l'}^r}\sum_{r'\in I_{k''}^{NIS}}\left(\frac{R_{k''r'l't-\theta}}{V_{k''r'l'}^{max}}\right) \leq 1 \quad \forall r\in E, l, l'\in L_r, \theta=0,..,c_{rl'l}, t\in H \tag{4.12}$$

$$\sum_{t=T-\tau_k+1}^{T} N_{klt} = 0 \quad \forall k \in K, l \in L_k \tag{4.13}$$

$$R^p_{krlt}, R^c_{krlt}, R_{krlt} \in \mathbb{R}_+ \quad \forall r \in I \cup P, k \in K^p_r, l \in L_r, t \in H$$
$$\xi_{klt} \in \mathbb{R}_+ \quad \forall r \in E, k \in K_r, l \in L_k, t \in H$$
$$\Pi^{slack}_{rld} \in \mathbb{R}_+ \;\; \forall r \in P, l \in L_r, d \in D_r \tag{4.14}$$
$$\Pi_{krlt} \in \mathbb{R}_- \quad \forall r \in I \cup P, l \in L_r, k \in K^p_r, t \in H$$
$$N_{klt} \in \{0,1\} \quad \forall\, r \in E, k \in K_r, l \in L_k, t \in H$$

Constraints (4.1) express the assignment of tasks to processing units and state that at most one task k of lot l can start during the time period corresponding to the task processing time. This is implemented through a backward time aggregation for $t' = t - \tau_k + 1$ over the binary variables N_{klt}. Since constraints (4.1) are similar to the STN constraints for handling task-unit allocation, M2 can be classified as a STN model.

Materials production R^p_{krlt} and consumption R^c_{krlt} are defined separately to address lots blending. Constraints (4.2) define the amount of resource r (intermediaries or final products) produced by task k of lot l at time interval t. Parameters $\nu^p_{kr\theta}$ give the production proportion of the batch size of task k for resource r. Constraints (4.3) give the amount of resource r consumed by task k of lot l at time interval t at the proportion $\nu^c_{kr\theta}$ of the batch size ξ_{klt}. Since resource r of lot l can be available from any tasks $k \in K^p_r$ that have produced r, the summation over R^c_{krlt} in the left-hand side of constraints (4.3) is required.

Constraints (4.4) define the special case of lots blending. In many situations, it is common to produce several lots of stable intermediaries that are used to produce other lots of final products. In these cases, blending of lots is allowed but it is necessary to ensure traceability, which is done by constraints (4.4). These constraints are defined for the set of intermediaries/products B whose lots can be blended.

Constraints (4.5) express the material r balance for each task k and lot l by considering the material in the previous time interval, the amount produced and consumed, and the material deliveries. Constraints (4.6) define the minimum and maximum materials/lots availability allowed for each time interval. Constraints (4.7) impose the task-batch size limits.

Constraints (4.8) define multiple product/lot deliveries Π_{krlt} for a given delivery time window DW_{rld}. The amount of resource r of lot l at delivery d is limited by the minimum Q_{rld}^{min} and maximum Q_{rld}^{max} quantities. Production requirements are modeled as "soft-constraints" so as to avoid infeasible solutions. Thus, missing deliveries are expressed by the continuous variables Π_{rld}^{slack} and are penalized in the objective function through coefficient c_{rld}^{slack}. Constraints (4.9) and (4.10) express the fact that delivery variables Π_{krlt} cannot take values for the time intervals out of the delivery time window and for other resources than final products.

Constraints (4.11) define the temporary storage in the processing units and state that if the binary variable N_{klt} is equal to 1, then R_{krlt} must be equal to 0. In other words, no task k of any lot l can start in the processing unit r if unit r is temporarily storing material from any other task. Note that the second term of the left-hand side only occurs for tasks that produce intermediaries subject to the Non-Intermediate Storage (NIS) policy, defined by the set I_k^{NIS}. Constraints (4.12) extend constraints (4.11) to account for sequence-dependent changeovers. In this way, tasks must respect the sequence-dependent changeover time defined for each unit and lot by the parameter $c_{rl'l}$ and for possible storage time in the processing units. Therefore, if task *k* of lot *l* occurs at time *t*, then the first term of the constraint is equal to one, and the second and third terms are forced to be zero for all tasks k' and k'' belonging to lots l' and for the time intervals corresponding to $t - \tau_k - \theta + 1$ for the production tasks and to $t - \theta$ for the temporary storage.

Constraints (4.13) define that tasks must finish in the time horizon of interest and constraints (4.14) state the non-negativity of the continuous variables resource availability, production and consumption, batch size, and missing delivery; the non-positivity of the delivery variables; and the integrality of the assignment/sequencing variables.

4.5.4 Objective Function

The objective is to maximize the economical result of the global operation (see expression (4.15)) by taking into account the value of the products (VP), the storage costs (SC), the operational costs (OC), and the missing deliveries costs (MC). Note that model M2 cannot take into account changeover costs since there are no changeover variables, and in order to make a fair comparison between models M1 and M2, changeover costs were not considered in the objective function.

Objective Function

$$\max Z = value\ of\ the\ products\ (VP) - storage\ costs (SC) \\ - operational\ costs\ (OC) \\ - missing\ delivery\ costs\ \ (MC) \tag{4.15}$$

$$VP = \sum_{r \in P} \sum_{l \in L_r} \sum_{k \in K_r^p} \sum_{t \in H} (-v_r \Pi_{krlt}) \tag{4.15a}$$

$$SC(M1) = \sum_{r \in I \cup P} \sum_{l \in L_r} \sum_{k \in K_r^{sto}} \sum_{t \in H} \left(c_r^{sto} (\xi_{klt} + R_{rlt})\right) \tag{4.15b}$$

$$SC(M2) = \sum_{r \in I \cup P} \sum_{l \in L_r} \sum_{k \in K_r} \sum_{t \in H} (c_r^{sto} R_{krlt}) \tag{4.15c}$$

$$OC = \sum_{k \in K_r} \sum_{l \in L_r} \sum_{t \in H} \left(c_k^{op} N_{klt}\right) \tag{4.15d}$$

$$MC = \sum_{r \in P} \sum_{l \in L_r} \sum_{d \in D_r} \left(c_{rld}^{slack} \Pi_{rld}^{slack}\right) \tag{4.15e}$$

The first term of the objective function defines the value of the delivered products, see expression (4.15a). The second term determines the storage costs, which are calculated differently for models M1 and M2. So, for model M1 storage costs are associated to materials stored under FIS and UIS policies that can be expressed by the continuous R_{rlt} variables and by materials temporarily held by the processing units, see expression (4.15b). For model M2, storage costs are determined simply by the continuous variables R_{krlt}, since the availability of the materials is only modeled through these variables, see expression (4.15c). The fourth term of the objective function determines the operational costs, see expression (4.15d) and the fifth term is a penalty cost associated with the missing deliveries, see expression (4.15e).

4.6 Models Extensions

We have also investigated some extensions of these models to address the start of changeovers tasks, non-preemptive lots, lots start and sizes, task-unit-layout assignment, and alternative task-unit assignment.

Changeovers Start (M1)

In model M1, as stated by constraints (4.A1), changeover tasks may occur in any time interval between the start of the tasks associated to lots l and l'. However, a common industrial practice is to perform the changeover as soon as the task finishes. We illustrate this situation in Figure 4.5, with a) showing the time range where the changeover tasks may occur if constraints (4.A1) are used. However, the desirable scheduling solution is the one presented in b), since the changeover occurs immediately after the storage tasks.

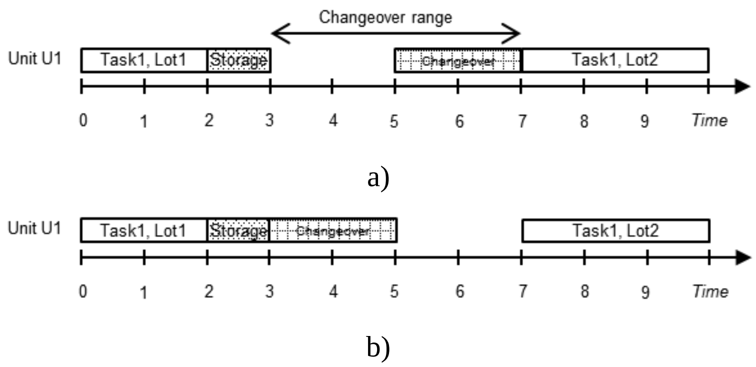


Figure 4.5 – Start of production, storage, and changeover tasks (Model M1).

Constraints (4.16) force changeovers to occur immediately after a production or storage task. Another relevant point is that constraints (4.16) help in reducing the model degeneracy.

$$\sum_{k \in K_r} N_{kl,t-\tau_k} - \sum_{l' \in L_r} C_{rll't} \geq 0 \quad \forall r \in E, l \in L_r, t \in H \tag{4.16}$$

Non - Preemptive lots (M1)

It is also a common practice in many chemical batch plants that lots once started in one unit cannot be interrupted to allow the production of a different lot. Constraints (4.17) define that if a changeover from lot l' to lot l occurs in unit r, then no changeover can occur in that unit from l to l'.

$$\sum_{l' \in L_r} \sum_{t \in H} C_{rl'lt} + \sum_{l' \in L_r} \sum_{t \in H} C_{rll't} \leq 1 \;\; \forall r \in E, l \in L_r \tag{4.17}$$

Lots Start (M1 and M2)

Constraints (4.18) state that lot l is only executed if the previous $l-1$ is also executed. Thus, if task k of lot l is performed at time t, then the same task k of lot $l-1$ should have started previously or any alternative tasks A_k to k should have started at the time intervals between $t' = 0$ and t. This allows different lots to be produced in parallel.

$$N_{klt} - \sum_{t'=0}^{t-\tau_k} N_{k,l-1,t'} - \sum_{k' \in A_k} \sum_{t'=0}^{t} N_{k',l-1,t'} \leq 0 \;\; \forall k \in K, l \in L_k, t \in H: l > 1 \tag{4.18}$$

Lots Sizes (M1 and M2)

If we want to define lots with exactly the same amount of material, constrains (4.19) and (4.20) may be applied. Constrains (4.19) impose that the total amount produced by tasks of different lots must be the same and constrains (4.20) state that the number of tasks must be the same among the lots.

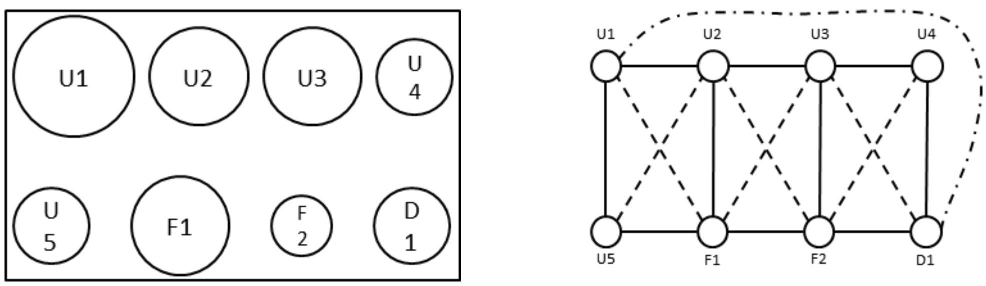
$$\sum_{t \in H} \xi_{klt} = \sum_{t \in H} \xi_{kl't} \qquad \forall k \in K, l, l' \in L_k, l \neq l', t \in H \tag{4.19}$$

$$\sum_{t \in H} N_{klt} = \sum_{t \in H} N_{kl't} \qquad \forall k \in K, l, l' \in L_k, l \neq l', t \in H \tag{4.20}$$

Task-Unit-Layout Assignment (M1 and M2)

For processes with many alternative units it may be preferable to do the task-unit assignment taking into consideration the physical layout of the units. Figure 4.6 depicts the plant layout and the allowed connections between units for the processes of Figure 4.1. For example, unit U1 can only transfer/receive materials to/from U2, U5, F1, and also D1.

This approach helps in the definition of physical aligned processes, leading in practice to several operational advantages.



a) b)

Figure 4.6 – Location of processing units: a) plant layout; b) allowable connection between units.

To model this requirement, we have created new binary variables X_{kl} that are equal to 1 if task *k* is assigned to lot *l*, see expression (4.21).

$$X_{kl} = \begin{cases} 1 & if\ task\ k\ is\ assigned\ to\ lot\ l \\ 0 & otherwise. \end{cases} \quad \forall\ k \in K, l \in L_k \tag{4.21}$$

If task k and k' use processing units that are connected, then $(k, k') \in S$. Constraints (4.22) define that if task k is assigned to lot l, then task k' cannot be assigned to the same lot, since *k* is not connected to *k'*. Constraints (4.23) ensure that if $\sum_{t \in H} N_{klt} > 0$ then $X_{kl} = 1$ and constraints (4.24) guarantee that if $\sum_{t \in H} N_{klt} = 0$, then $X_{kl} = 0$.

$$X_{kl} + X_{k'l} \leq 1 \quad \forall\ (k, k') \notin S, l \in L \tag{4.22}$$

$$\sum_{t \in H} N_{klt} \leq\ X_{kl} M \quad \forall\ k \in K, l \in L_k \tag{4.23}$$

$$\sum_{t \in H} N_{klt} \geq\ X_{kl} \quad \forall\ k \in K, l \in L_k \tag{4.24}$$

Alternative Task-Unit Assignment (M1 and M2)

Moreover, we may want to ensure that from the alternative units available for each task only one is assigned. Constraints (4.25) guarantee that from the alternative tasks A_k to *k* only one is selected.

$$X_{kl} + \sum_{k' \in A_k} X_{k'l} \leq 1 \quad \forall k \in K, l \in L_k \tag{4.25}$$

4.7 Numerical Results

In order to show how general model M2 is and to compare its effectiveness, we consider four different chemical processes. Process 1 was firstly addressed by Kondili et al. (1993), Process 2 was published by Kallrath (2002), Process 3 was proposed in the paper of Papageorgiou and Pantelides (1993) and finally Process 4, depicted in Figure 4.1, is proposed by us. The first three processes are benchmark problems from the literature and fairly represent the existing scheduling complexities of the multipurpose batch plants. The last process is intended to allow an analysis of lots blending and traceability features and the model extensions.

We present the solution statistics (integer and continuous variables, nodes, iterations, linear relaxation at the root node, integrality gap, objective function value, and CPU time) of models M1 and M2 for four scheduling horizons (24, 48, 120 and 240 hours) and for different time grids, whenever this is applicable.

Model M1 is defined by constraints (4.A3) to (4.A12) if changeovers are not present, and by constraints (4.A1), (4.A2), and (4.A4) to (4.A12) if changeovers are modeled. Model M2 is defined by constraints (4.1) to (4.11), (4.13), and (4.14) if changeovers are not required; and by constraints (4.1) to (4.10), and (4.12) to (4.14) if changeovers are needed. The objective function is to maximize the economical result of the global operation and is the same for both models, despite the modeling differences in the storage costs discussed in section 4.5.4.

The models were implemented using ILOG/CPLEX version 12.5, running on an Intel Xeon X5680 at 3.33GHz with 24 GB of RAM. We have considered the time limit of 3,600 seconds and the integrality gap of 5% as stopping criteria, so as to evaluate the models performance respecting the time to obtain solutions and their quality. The networks of processes P1, P2, and P3 and respective data tables are given in the supporting information.

4.7.1 Process 1

Process 1 is the network published by Kondili et al. (1993). This process involves a cyclic material flow, alternative processing units, and different storage policies. Additionally, we have performed a slight modification of the network by considering the NIS policy for the intermediaries HOTA, INTBC, and IMPE. Because Process 1 has a unique network, no changeovers were defined. Moreover, we assume a single lot, thus lots blending are not considered and materials traceability is implicitly ensured.

Numerical results for Process 1 are depicted in Table 4.1 for the case where the stopping criterion is the time limit equal to 3,600 seconds and in Table 4.2 where the stopping criterion is the integrality gap of 5%. As expected, model M2 always has less binary variables and more continuous variables and constraints when compared with model M1. This is because M1 makes use of binary variables to model storage tasks, while M2 implements storage through the set of constraints (4.11).

For the 24 hours scheduling horizon both models proved optimality relatively fast. However, in the 48 hours instance, the solution time of M2 is lower than the time required by M1 to prove optimality. The same happens when trying to obtain a solution within the margin of 5% of the integrality gap.

In the 120 and 240 hours instances none of the models succeeded to prove optimality. In the horizon of 120 hours, M1 is slightly better than M2, and in the 240 hours instance, the solution of M2 is better than M1. Assuming a margin of 5% for the integrality gap, the 120 hours instance of M1 reached a solution in 201 seconds, while M2 took 381 seconds. However in the 240 hours instance, M2 reached a better solution in just 460 seconds, while M1 required 2,053 seconds.

Globally, model M2 ran very well and outperformed model M1 in most of the instances.

Table 4.1 – Process 1 solution statistics (stopping criterion is the time limit of 3,600 seconds).

Model/process/ horizon/grid	Int. variables/ Cont. variables/ Constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
M1/P1/24/1	325/732/1334	5,331	314,722	30,673.1	0.00	28,709.6	3
M2/P1/24/1	200/1303/1635	6,278	310,912	31,389.1	0.00	28,709.6	2
M1/P1/48/1	637/1430/2607	149,480	17,765,652	62,828.7	0.00	60,380.9	287
M2/P1/48/1	392/2553/3196	118,801	9,257,543	63,033.6	0.00	60,380.9	189
M1/P1/120/1	1,573/3520/6472	354,705	37,582,562	152,026.4	1.05	148,434.4	3,600
M2/P1/120/1	968/6299/7925	639,012	69,419,840	152,115.5	1.20	148,295.2	3,600
M1/P1/240/1	3133/7002/12929	111,232	19,898,360	290,651.7	2.06	283,334.0	3,600
M2/P1/240/1	1928/12541/15822	203,522	37,267,461	290,570.4	1.46	285,068.8	3,600

Table 4.2 – Process 1 solution statistics (stopping criterion is the integrality gap of 5%).

Model/process/ horizon/grid	Nodes	Iterations	Gap (%)	Objective	CPU time (sec)
M1/P1/48/1	12,084	1,706,400	5.00	59,090.5	32
M2/P1/48/1	9,562	953,946	4.81	59,427.9	17
M1/P1/120/1	19,637	2,160,873	4.33	144,776.6	201
M2/P1/120/1	55,064	8,342,967	4.28	144,811.8	381
M1/P1/240/1	53,307	9,155,148	4.15	277,682.5	2,053
M2/P1/240/1	45,036	5,870,111	3.24	280,300.0	460

4.7.2 Process 2

Process 2 was published by Kallrath (2002) and is being extensively used as a benchmark problem because of its complexity. The process suggested by the author accounts for flexible output proportions for intermediaries, several storage policies, a cyclic material flow, and a considerable number of states, units, and tasks. In this paper, we do not consider flexible output proportions; therefore, the proportion of material going to State3 was fixed to 0.3 and the proportion of material going to State4 was fixed to 0.7. Again, we assume a single lot and that there are no sequence-dependent changeovers.

The solution statistics presented in Table 4.3 show that model M2 performed better than model M1 in all instances. In the 48 hours horizon, M1 proved optimality in 233 seconds, while M2 just took 101 seconds. With the increase of the model size, both models had difficulties in reaching an optimal solution; however, M2 obtained always the best solution within the specified time limit. In the 120 hours horizon, the solution obtained by M1 was within a gap of 12.86%, while the solution retrieved by M2 ensured a gap of 3.11%.

In this process, we have opted not to test the stopping criterion of the 5% of integrality gap, because the larger instances showed to be very hard to solve with both models.

Table 4.3 – Process 2 solution statistics (stopping criterion is the time limit of 3,600 seconds).

Model/process/ horizon/grid	Int. variables/ Cont. variables/ Constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
M1/P2/48/1	1421/3632/6759	29,345	93,866,983	5,269.3	0.00	4,802.8	233
M2/P2/48/1	1176/6278/8577	20,788	3,992,399	5,247.9	0.00	4,802.8	101
M1/P2/120/1	3509/8965/16696	15,464	26,488,007	16,725.8	12.86	14,495.0	3,600
M2/P2/120/1	2904/15499/21178	55,076	21,512,065	16,611.9	3.11	15,440.1	3,600
M1/P2/240/1	6989/17850/33305	1,088	8,328,162	33,261.4	17.37	27,734.6	3,600
M2/P2/240/1	5784/30864/42227	14,876	12,549,594	32,925.6	11.59	28,519.4	3,600

4.7.3 Process 3

Process 3 is from Papageorgiou and Pantelides (1993) and is defined by three parallel production lines that share almost all processing units. The processes have several storage policies, including ZW and NIS, and have tasks with small and large processing times. Here, we consider sequence-dependent changeovers between products that have a single lot, and we test these processes with time grids of one and five hours.

In Table 4.4, we show the results with a time grid of 5 hours and for scheduling horizons of 120 and 240 hours. It is not possible to run Process 3 for smaller time horizons, because tasks have large processing times. Model M2 outperformed model M1 in both instances. In the 120 hours horizon, M1 proved optimality in 5 seconds, which required 18 seconds. And in the 240 hours horizon, M2 proved optimality in just 476 seconds, while M1 needed 677 seconds. The number of nodes and iterations of the branch-and-bound for model M2 are also significantly smaller when compared with those of model M1. Considering the stopping criterion of 5% in the integrality gap, see Table 4.5, model M1 obtained a solution in just 10 seconds, while M2 required 58 seconds.

By assuming a time grid of 1 hour, the model size naturally increased in a significant way and none of the models proved optimality, see Table 4.6 and Table 4.7. M2 performed better than M1 in all instances, always reaching an integrality gap within 5%, with the exception of one instance

In this process, model M2 had better performance in all indicators, suggesting that model M2 works well in instances having multiple processes, with sequence-dependent changeovers and different storage policies.

Table 4.4 – Process 3 solution statistics (time grid is 5 hours).

Model/process/ horizon/grid	Int. variables/ Cont. variables/ Constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
M1/P3/120/5	1550/1720/2841	17,969	1,413,603	5,387.3	0.00	5,066.0	18
M2/P3/120/5	575/2879/5086	5,893	297,682	5,392.2	0.00	5,066.0	5
M1/P3/240/5	3038/3355/5541	330,548	40,797,510	10,774.7	0.00	10,184.8	677
M2/P3/240/5	1127/5642/9946	199,201	21,796,427	10,784.5	0.00	10,184.8	476

Table 4.5 – Process 3 solution statistics (time grid is 5 hours and stopping criterion is the integrality gap of 5%).

Model/process/ horizon/grid	Nodes	Iterations	Gap (%)	Objective	CPU time (sec)
M1/P3/240/5	18,539	2,565,141	4.30	10,000.6	58
M2/P3/240/5	4,771	446,278	5.00	9,966.8	10

Table 4.6 – Process 3 solution statistics (time grid is 1 hour and stopping criterion is the time limit of 3,600 seconds).

Model/process/ horizon/grid	Int. variables/ Cont. variables/ Constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
M1/P3/120/1	7502/8248/13653	178,586	72,707,311	5,317.2	13.34	4,587.7	3,600
M2/P3/120/1	2783/13919/30346	168,092	25,231,890	5,321.2	9.09	4,768.5	3,600
M1/P3/240/1	14942/16411/27165	56,025	26,545,793	10,634.3	11.18	9,457.9	3,600
M2/P3/240/1	5543/27722/60418	59,941	17,039,145	10,642.4	5.80	9,920.5	3,600

Table 4.7 – Process 3 solution statistics (time grid is 1 hour and stopping criterion is the integrality gap of 5%).

Model/process/ horizon/grid	Nodes	Iterations	Gap (%)	Objective	CPU time (sec)
M1/P3/120/1	578,271	296,170,781	7.84	4,795.6[1)]	14,400
M2/P3/120/1	566,936	131,212,184	4.78	4,912.4	14,400
M1/P3/240/1	341,515	136,133,027	10.24	9,496.7[1)]	14,400
M2/P3/240/1	129,924	44,592,826	4.98	9,978.7	7,125

[1)] Stopping criterion is the time limit of 14,400 seconds.

4.7.4 Process 4

We now consider the network defined by the three processes depicted in Figure 4.1. Products PA and PB are produced from raw materials, while Product PC is produced from PA and PB. Moreover, we are given the unit's physical layout shown in Figure 4.6. This process is used to test the performance of both models and also to address new modeling features only possible to be treated with model M2 with the extensions proposed in section 4.6.

First, we test Process 4 assuming sequence-dependent changeovers and single lots without blending. Since model M1 cannot address lots blending and traceability features, we slightly change the recipe of product PC by imposing that the materials required to produce PC are raw materials and not the products PA and PB as defined in Figure 4.1. The numerical results for this scenario are presented in Table 4.8 and Table 4.9. Second, we define multiple lots and assume that lots blending may happen. Thus, here only model M2 is tested. We analyze lots traceability, sequence-dependent changeovers, temporary storage in the processing units, task-unit-layout, and alternative task-unit assignments. The numerical results for this case are shown in Table 4.12.

Single Lots Without Blending

As it can be seen in Table 4.8 and Table 4.9, the results obtained by model M2 are superior to the results retrieved by model M1. For example, in the 48 hours horizon instance, the solution time of M2 is 651 seconds, while M1 required 1,996 seconds.

In the 120 and 240 hours horizons, none of models could prove optimality for the CPU time limit of 3,600 seconds. Nevertheless, the solutions obtained by M2 are always better, achieving integrality gaps that are less than 5% and that are less than half of the gaps obtained by M1.

Table 4.8 – Process 4 solution statistics (stopping criterion is the time limit of 3,600 seconds).

Model/process/horizon/grid	Int. variables/cont. variables/ constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
M1/P4/24/1	1750/1771/3103	9,259	607,242	515,216.8	0.01	511,167.8	9
M2/P4/24/1	500/2704/6331	1,945	84,077	515,295.8	0.01	511,167.8	6
M1/P4/48/1	3430/3454/6064	315,067	91,541,309	1,030,433.6	0.01	1,022,336.5	1,996
M2/P4/48/1	980/5299/12388	72,764	5,372,800	1,030,591.7	0.01	1,022,336.5	651
M1/P4/120/1	8470/8497/15025	53,326	18,026,661	2,545,650.8	4.25	2,430,807.0	3,600
M2/P4/120/1	2420/13078/30637	89,081	13,183,238	2,548,144.6	1.27	2,499,456.8	3,600
M1/P4/240/1	16870/16900/29986	20,785	17,080,636	5,008,652.8	8.96	4,572,396.1	3,600
M2/P4/240/1	4820/26041/61078	46,029	12,090,984	5,015,842.1	3.52	4,809,863.8	3,600

With an integrality gap of 5% as stopping criterion, model M2 also performed better than M1, as it can be seen in Table 4.9. The solution times of M2 are considerably smaller than the solution times required by M1 (except for the 24 hours instance). In the 120 hours horizon, M1 required 2,554 seconds and M2 required 51 seconds, and in the 240 hours instance M1 needed 10,370 seconds, while M2 just needed 267 seconds.

Table 4.9 – Process 4 solution statistics (stopping criterion is the integrality gap of 5%).

Model/process /horizon/grid	Nodes	Iterations	Gap (%)	Objective	CPU time (sec)
M1/P4/24/1	76	26,378	0.52	511,067.8	3
M2/P4/24/1	287	24,182	0.59	511,075.8	6
M1/P4/48/1	12,359	3,340,246	4.14	985,409.0	91
M2/P4/48/1	2,055	150,354	3.71	991,374.5	14
M1/P4/120/1	53,308	18,026,661	4.39	2,427,689.0	2,554
M2/P4/120/1	1,762	269,770	3.99	2,441,242.8	51
M1/P4/240/1	55,073	43,710,694	4.27	4,775,346.8	10,370
M2/P4/240/1	6,889	931,290	3.90	4,806,198.8	267

The performance of the extensions, on changeovers start and non-preemptive lots, expressed by constraints (4.16) and (4.17), respectively, is assessed by the numerical results of Table 4.10. Constraints (4.17) impose that lots cannot be interrupted to produce other lots, thus limiting the profit of the schedule when compared with the profit values shown in Table 4.8. The computational performance of the model M1.1 tends to decrease with the increase of the time horizon, as can be seen by the large integrality gaps of the 120 and 240 hours instances.

Table 4.10 – Process 4 solution statistics, assuming changeovers start and non-preemptive lots constraints.

Model/process/ horizon/grid	Constraints	Gap (%)	Objective	CPU time (sec)
M1.1/P4/24/1	3,545	0.00	501,290.0	10
M1.1/P4/48/1	6,914	0.57[1)]	998,965.5	3,600
M1.1/P4/120/1	17,099	23.02[1)]	2,060,248.0	3,600
M1.1/P4/240/1	34,100	34.94[1)]	3,692,146.0	3,600

[1)] Stopping criterion is the time limit of 3,600 seconds.

On the Changeover Costs

In order to reflect the changeover costs on the schedule solutions, we have added expression (4.26) in the objective function of model M1. The cost structures of the resultant model M1.2 and of model M2 are illustrated in Figure 4.7.

$$GC = \sum_{r \in E} \sum_{l \in L_r} \sum_{l' \in L_r} \sum_{t \in H} (c_{rll'} C_{rll't}) \tag{4.26}$$

It can be seen that in the 24 hours instance both models had storage costs equal to 3,032 m.u. In the 48 hours storage costs increased to 6,463 in model M2, while model M1.2 storage costs (SC) increased to 6,807. Regarding the operational costs (OP), M2 had always inferior costs than M1.2. It is important to note that, when changeover costs (GC) are considered in the objective function the tradeoffs between task-unit allocation, storage and changeover costs pass to exist.

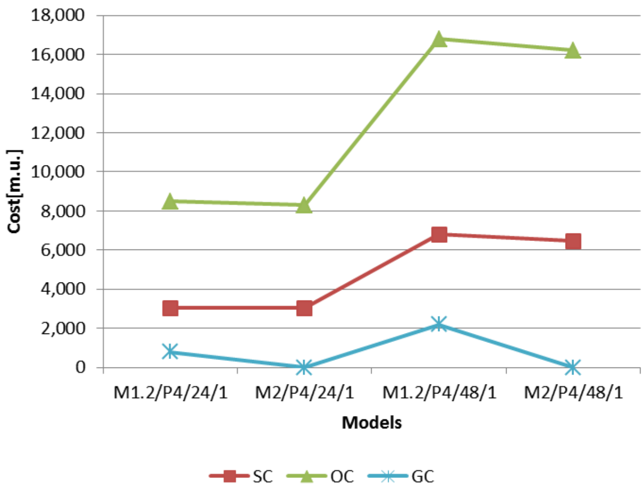


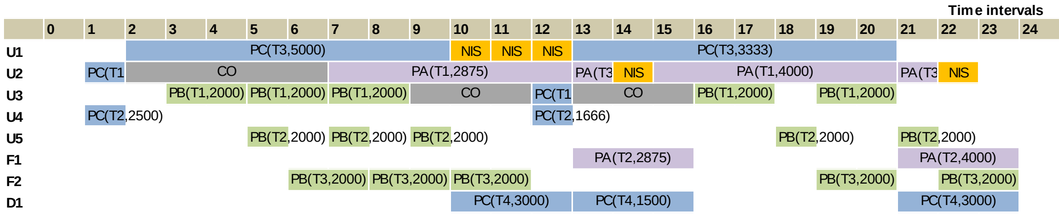
Figure 4.7 – Cost structure for models M1.2 and M2 (SC – storage costs, OC – operational costs, GC – changeover costs).

The computational results of model M1.2 are shown in Table 4.11. As expected, the profit obtained by M1.2 is always inferior to the profit obtained by models M1 and M2 (see Table 4.8) due to the changeover costs. Model M1.2 demonstrated worse performance than M1, particularly in the larger instances. For example, in the 240 hours instance, M1.2 had 28.23% of integrality gap, in contrast to M1 that had 8.96%.

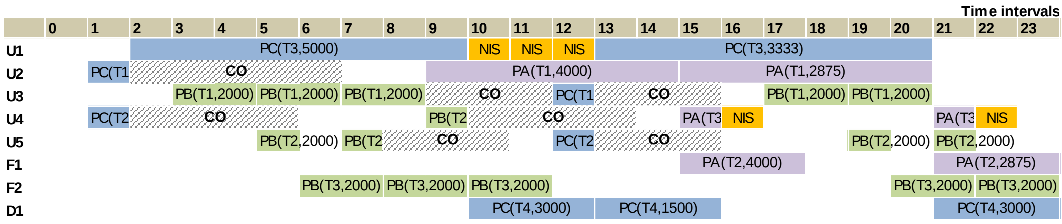
Table 4.11 – Process 4 solution statistics, assuming changeovers costs.

Model/process/ horizon/grid	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
M1.2/P4/24/1	4,291	476,489	515,200.6	0.01	510,167.8	8
M1.2/P4/48/1	300,486	130,181,404	1,030,401.2	0.11	1,019,193.5	3,600
M1.2/P4/120/1	52,633	35,118,731	2,545,454.9	5.14	2,408,360.0	3,600
M1.2/P4/240/1	12,733	12,292,104	5,007,848.3	28.23	3,884,643.0	3,600

Looking into the scheduling solutions (see Figure 4.8), we can analyze how changeover costs affect the task-unit assignment. The schedule solution of M1.2 has a total of 3 changeovers, resulting into a cost of 800 m.u. and an idle time of 11 hours. Although M2 does not model changeover tasks, costs and time of the changeovers can be derived by analyzing the schedule solution. In this way, the schedule solution of M2 has a total of 7 changeovers that result into a cost of 1800 m.u. and an idle time of 25 hours. Processing units are used less efficiently in M2, which concerns to the total changeover time and costs. Nevertheless, the profit of M2 is 99.8% of M1.2, discounting the changeover cost of 1800 m.u. to the profit obtained by M2. Thus, although M1.2 and M2 schedules are slightly different, they deliver the same amount of products and have a similar profit. In practice, since changeover constraints lead to a more efficient model, they can be used instead of changeover tasks if: a) the exact time of the changeover is not relevant; b) utilities/materials consumption during changeovers can be disregarded; and c) changeover costs are not significant.



a)



b)

Figure 4.8 – Scheduling for 24 hours instance: a) model M1.2; b) model M2. (CO = changeover)

Multiple Lots with Blending

Now, we use the processes as shown in Figure 4.1 to obtain schedules with multiple lots per product and with blending operations. We considered the production of two lots of PA, two lots of PB, and a single lot of PC, in a time horizon of 48 hours. The aim is to define production schedules in which the traceability of lots is kept during the entire horizon and the tasks-units assignment is done by assuming the physical layout limitations shown in Figure 4.6. For that we consider the extended model M2.1 by adding constraints (4.22) to (4.25) to model M2. Moreover, we also include the lot sizes extensions in model M2.2 that is defined by constraints (4.19) to (4.25).

Figure 4.9 shows the schedule for a 48 hours horizon, having an objective value of 340,442.2 m.u., relative to a delivery of 7,000 kg of product PC. Lot L1 of product PA starts first in units U2 and F1, while lot L2 of the same product is processed in units U4 and F2. We can see that the physical layout limitations expressed in Figure 4.6 were followed by both lots. Regarding the production of PB, lots L1 and L2 were produced in units U3, U3, and F2.

Although model M2.1 does not explicitly give the start of the temporary storage tasks in the processing units and the sequence-dependent changeovers, those can be directly deduced from R_{krlt} and N_{klt} variables. Thus, it can be seen that intermediary PA_S3 is temporary stored in unit U3 in all occurrences of TASK3 of product PA.

Concerning the sequence-dependent changeovers, we can see changeovers between lots of different products and changeovers between lots of the same product. This latter case happens in unit U3 at the time interval 9.

Finally, lots traceability is ensured for all products. The amounts produced of PA and PB of each lot are consumed by product PC and are directly traceable. For example, at the time interval 27, the amount of lot L1 of product PB is 2,500 kg and of L2 is 833.3 kg, and because the amount of L1 of PB is not sufficient to feed the batch of TASK2 of PC, it is necessary to blend lots. This situation can be seen in Figure 4.10, at time 28, where lots L1 and L2 of PB are consumed simultaneously by TASK2 of product PC.

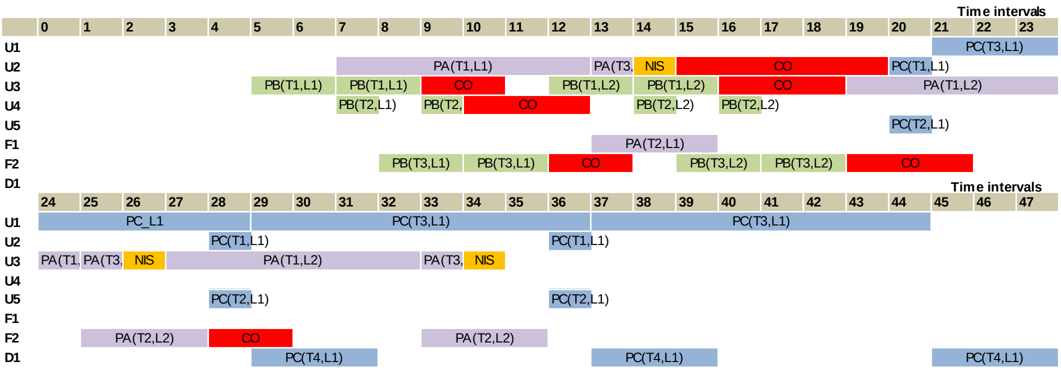


Figure 4.9 – Scheduling for instance M2.1/P4/48/1[1)].

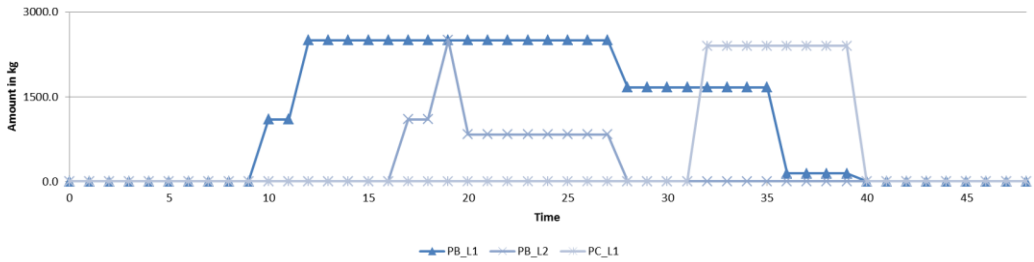


Figure 4.10 – Inventory for lots of products PB and PC.

Table 4.12 shows the computational results for models M2.1 and M2.2. M2.1 obtained a profit equal to 340,442.2 after 3,600 seconds. But assuming an integrality gap of 5%, a solution was retrieved in just 31 seconds. Model M2.2 takes into account the lot sizes constraints (4.19) and (4.20), and it can be seen that M2.2 and M2.1 performances are comparable for the tested instance.

Table 4.12 – Process 4 solution statistics.

Model/process/ horizon/grid	Int. variables/cont. variables/ constraints	Nodes	Iterations	Gap (%)	Objective	CPU time (sec)
M2.1/P4/48/1[1)]	1650/8875/30554	36,169	31,493,851	0.61	340,442.2	3,600
M2.1/P4/48/1[2)]	1650/8875/30554	935	233,990	2.36	338,356.4	31
M2.2/P4/48/1[1)]	1650/8875/30580	57,805	34,439,408	0.14	338,787.2	3,600
M2.2/P4/48/1[2)]	1650/8875/30580	236	181,674	4.13	332,480.2	34

[1)] Stopping criterion is the time limit of 3,600 seconds;

[2)] Stopping criterion is the integrality gap of 5%.

4.8 Conclusions

In this work, we propose two general discrete-time scheduling models for multipurpose batch plants (models M1 and M2). We first use a RTN discrete-time formulation (M1) as basis for comparison with a more innovative model (M2). The first model (Model M1) extends the RTN model of Pantelides (1994) by considering explicitly and in an integrated way scheduling features already treated in the literature, such as temporary storage in the processing units (Kondili et al., 1993), sequence-dependent changeovers (Castro, Novais, et al., 2008), and multiproduct delivery extensions (Wassick & Ferrio, 2011). This model is then generalized by considering the start of changeovers tasks, non-preemptive lots, as well as alternative task-unit and task-unit-layout assignments.

Model M2, based on STN, can be viewed as an innovative contribution in the area, explicitly modeling the inventory carried out in each task by adding a task index to the resource availability variables. This approach allows the development of new types of constraints for modeling sequence-dependent changeovers and temporary storage in the processing units. Moreover, we address lots blending, lots start, and alternative task-unit and task-unit-layout assignments. Lots blending and traceability are two requirements introduced in this work that are common in the chemical and biochemical-pharmaceutical industries, considered here with the purpose of keeping record of the blending processes during the production.

We compare the effectiveness of both models using three benchmark problems from the literature and one scheduling problem proposed in this paper. Experimental results have shown that model M2 is computationally more effective for the instances tested. In the larger or more complicated instances, both models had difficulties in proving optimality. However, model M2 always reached a solution within 5% of the integrality gap, except for the 240 hours scheduling horizon of Kallrath (2002) network. Model M1 had worse performance in most of the cases.

Two critical modeling features of the discrete-time formulations (sequence-dependent changeovers and temporary storage in the processing units) have been addressed, the proposed modeling alternative being computationally more efficient. An interesting and challenging issue for future research is the modeling of variable processing times with discrete-time formulations.

Appendix - RTN Model (M1)

$$R_{rlt} = \left(R_{rl}^{init}|_{(t=0)}, R_{rl,t-1}|_{t>0}\right) + \sum_{k \in K_r} \sum_{\theta=0}^{\tau_k} \left(\mu_{kr\theta} N_{kl,t-\theta}\right) + \sum_{l' \in L_r} \sum_{l'' \in L_r} \sum_{\theta=0}^{c_{rl'l''}} \left(\alpha_{rl'l''l\theta} C_{rl'l'',t-\theta}\right) \quad \forall r \in E, l \in L_r, t \in H \tag{4.A1}$$

$$\sum_{l \in L_r} R_{rl}^{init} \leq 1 \quad \forall r \in E \tag{4.A2}$$

$$R_{rt} = \left(R_{r}^{init}|_{(t=0)}, R_{r,t-1}|_{t>0}\right) + \sum_{k \in K_r} \sum_{\theta=0}^{\tau_k} \left(\mu_{kr\theta} N_{kl,t-\theta}\right) \quad \forall r \in E, l \in L_r, t \in H \tag{4.A3}$$

$$R_{rlt} = \left(R_{rl}^{init(m)}|_{(t=0)}, R_{rl,t-1}|_{t>0}\right) + \sum_{k \in K_r} \sum_{\theta=0}^{\tau_k} \left(\nu_{kr\theta} \xi_{kl,t-\theta}\right) + \Pi_{rlt} \quad \forall r \in I \cup P, l \in L_r, t \in H \tag{4.A4}$$

$$0 \leq \sum_{l \in L_r} R_{rlt} \leq R_{rt}^{max} \quad \forall r \in I \cup P, t \in H \tag{4.A5}$$

$$V_{krl}^{min} N_{klt} \leq \xi_{klt} \leq V_{krl}^{max} N_{klt} \quad \forall\, r \in E, k \in K_r, l \in L_k, t \in H \tag{4.A6}$$

$$\xi_{k'lt} \leq \xi_{k'l,t-1} + \sum_{\theta=0}^{\tau_k} \nu_{kr\theta}^{p} \xi_{kl,t-\theta} \quad \forall r \in I^{NIS}, k' \in K_r^{sto}, k \in K_r^{p}, l \in L_r, t \in H \tag{4.A7}$$

$$Q_{rld}^{min} - \Pi_{rld}^{slack} \leq \sum_{t \in DW_{rld}} (-\Pi_{rlt}) \leq Q_{rld}^{max} \quad \forall r \in P, l \in L_r, d \in D_r$$
$$DW_{rld} = \{t \mid r \in P, l \in L_r, d \in D_r, t \in H : T_{rd}^{dd} \geq t \geq T_{rd}^{ed}\} \tag{4.A8}$$

$$\Pi_{rlt} = 0 \quad \forall r \in P, l \in L_r, d \in D_r, t \in H \backslash DW_{rld} \tag{4.A9}$$

$$\Pi_{rlt} = 0 \quad \forall r \in I, L \in L_r, t \in H \tag{4.A10}$$

$$\sum_{t=T-\tau_k+1}^{T} N_{klt} = 0 \quad \forall k \in K, l \in L_k \tag{4.A11}$$

$$\begin{gathered} R_{rlt} \in \mathbb{R}_+ \quad \forall r \in E \cup I \cup P, l \in L_r, t \in H \\ \xi_{klt} \in \mathbb{R}_+ \quad \forall r \in E, k \in K_r, l \in L_k, t \in H \\ \Pi_{rld}^{slack} \in \mathbb{R}_+ \quad \forall r \in P, l \in L_r, d \in D_r \\ \Pi_{rlt} \in \mathbb{R}_- \quad \forall r \in I \cup P, l \in L_r, t \in H \\ N_{klt} \in \{0,1\} \quad \forall\, r \in E, k \in K_r, l \in L_k, t \in H \\ C_{rl'l''t} \in \{0,1\} \quad \forall r \in E, l', l'' \in L_r, t \in H \end{gathered} \tag{4.A12}$$

Constraints (4.A1) express the availability of the processing units for each lot and time interval. So the unit availability R_{rlt} is equal to the availability in the previous time interval $R_{rl,t-1}$ plus the availability resulting from the unit's allocation/release to/from the production or changeover tasks at time interval *t*. For the production tasks, this is done through coefficient $\mu_{kr\theta}$ that defines the unit *r* allocation/release done by task *k* at time θ relative to the start of the task. And for changeover tasks, we have introduced the changeover coefficient $\alpha_{rl'l''l\theta}$ that defines the allocation/release of unit *r* from lot *l'* to *l''*, at the current lot *l* and at time θ relative to the start of the changeover task. The changeover time is given by parameter $c_{rl'l''}$. Constraints (4.A2) do the initial assignment of processing units to lots. A simplified version of constraints (4.A1) can be written if changeovers between lots are not required; see constraints (4.A3). In these cases, we just have the resource balance for the production tasks and the index *l* of the resource availability variables is removed. Because constraints (4.A1) or (4.A3) ensure that no processing units are eliminated or created, we do not need to define lower or upper bounds for this type of resources. Note that, R_{rlt} or R_{rt} variables do not need to be integer variables, since the resource balance equation ensures that these variables take always integer values.

The materials balance constraints (4.A4) are similar to the units balance constrains (4.A1) or (4.A3). The difference is that constraints (4.A4) handle intermediaries and final products and not processing units. Materials are consumed and produced at the proportion $\nu_{kr\theta}$ of the batch size ξ_{klt}. The continuous variables Π_{rlt} express the deliveries of product *r* of lot *l* at the time interval *t* and will always have non positive values; thus, no material receipts are expected to occur during the scheduling horizon. We opted not to model raw materials since it can be assumed, without loss of generality, that raw materials are available when needed. Constraints (4.A5) define the minimum and maximum materials availability allowed for each time interval. These constraints also permit the definition of different storage policies depending on the value of parameters R_{rt}^{max}. Thus, R_{rt}^{max} take the value 0 for Non-Intermediate Storage (NIS) or Zero-wait (ZW) and take a value greater than zero if there is Finite Intermediate Storage (FIS) or Unlimited Intermediate Storage (UIS). In the latter case the value should be sufficiently large to account for unlimited storage capacity. Constraints (4.A6) define that the batch size ξ_{klt} must be within the minimum V_{krl}^{min} and maximum V_{krl}^{max} allowed capacities of resource *r* and task *k* of lot *l*.

Constraints (4.A7) were first proposed by Kondili et al. (1993) to model temporary storage done by the processing units (NIS policy) and ensure that the intermediary is held by the unit in which it was produced. These constraints require the creation of additional storage tasks to model the NIS policy and impose that the batch size of a storage task is less than or equal to the previous amount stored plus the amount produced at each time interval. If the batch size of a storage task is greater than zero, then the assignment binary variable for the storage task must be one by constraints (4.A6). Parameters $\nu_{kr\theta}^{p}$ give the production proportion of the batch size of task k for resource *r,* and I^{NIS} is a subset of I that has the intermediaries subject to the NIS policy. Note that storage tasks have duration equal to one since materials availability needs to be checked at every time interval. These constraints are only required in the cases that the alternative units suitable to perform a given task are dissimilar. In these situations, constraints (4.A7) guarantee that the unit allocated during the storage period is the same unit that has produced the material being held.

Multiple product deliveries are defined by constraints (4.A8). The delivery time windows DW_{rld} are defined by fixed time intervals in which the product deliveries can happen. Constraints (4.A9) and (4.A10) set the delivery variables to zero for the time intervals out of the delivery time window and for other resources rather than final products.

Constraints (4.A11) define that tasks must finish in the time horizon of interest. Finally, constraints (4.A12) guarantee the non-negativity of the continuous variables resource availability, batch size, and missing deliveries; the non-positivity of the delivery variables; and the integrality of the assignment/sequencing variables and sequence-dependent changeovers.

Supporting Information

Networks

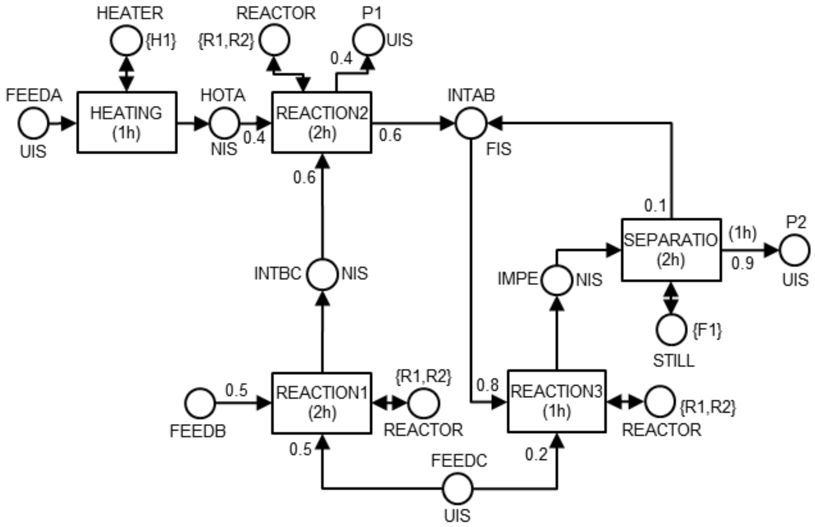


Figure 4.S1 – Process 1, Kondili et al. (1993).

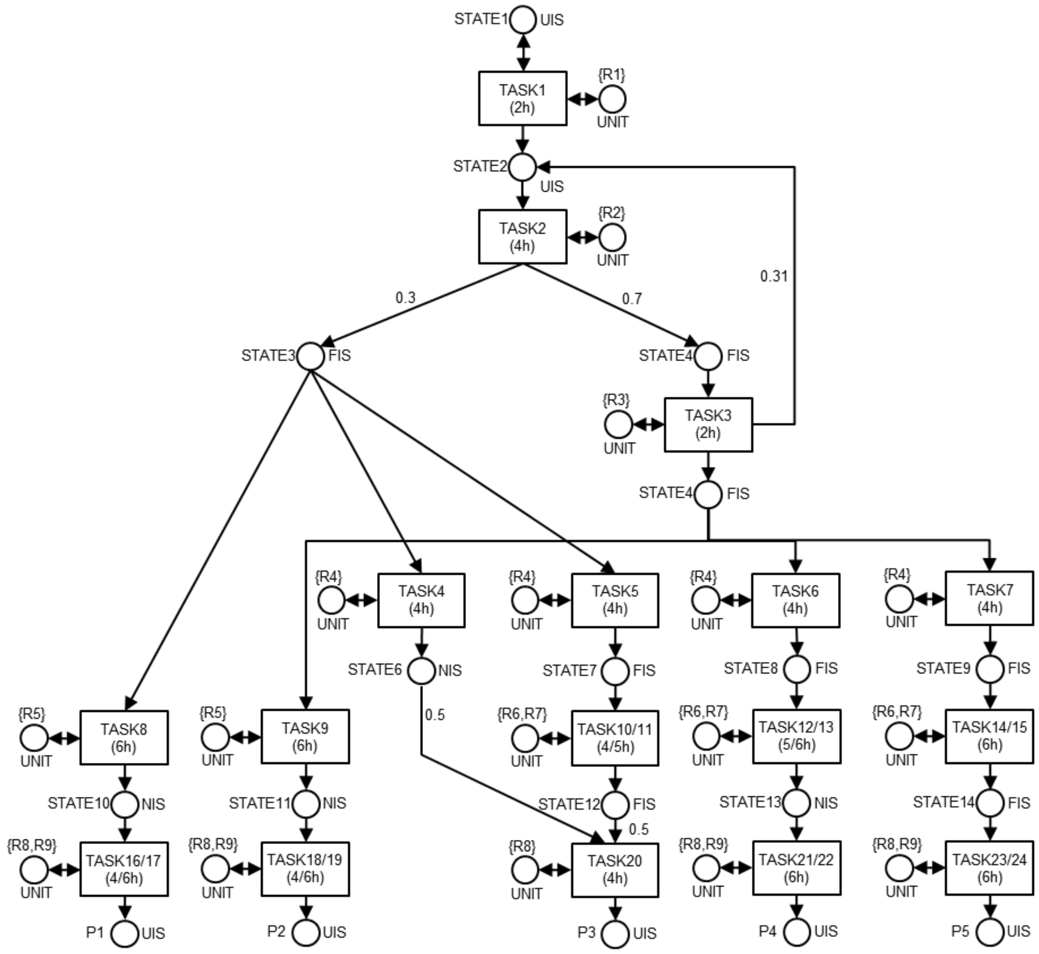


Figure 4.S2 – Process 2, Kallrath (2002).

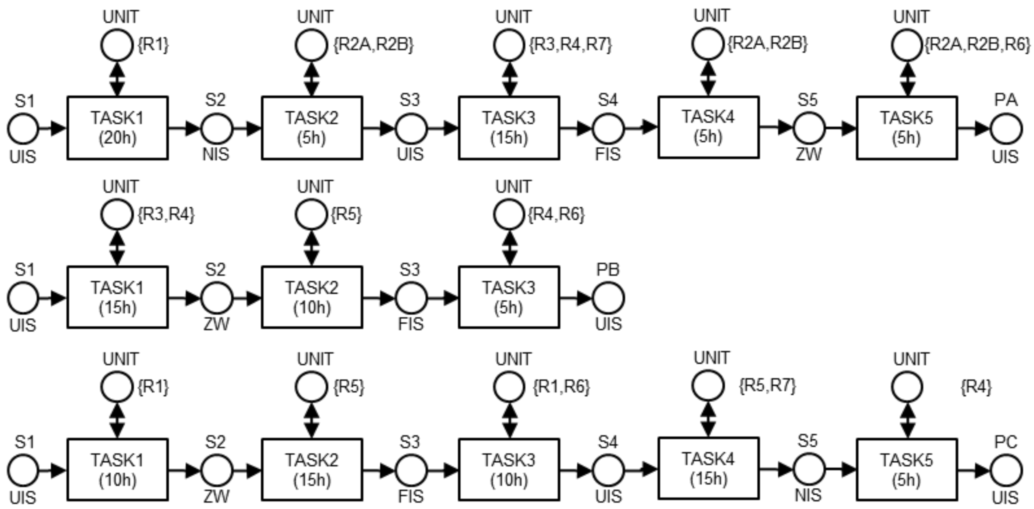


Figure 4.S3 – Process 3, Papageorgiou and Pantelides (1993).

Data

Table 4.S1 – Process 1: products demand, delivery dates, price and miss deliveries costs.

Network	Product	Lot Product	Earliest delivery Date	Latest delivery Date	Minimum lot size [kg]	Maximum lot size [kg]	Price [m.u./kg]	Miss deliveries costs [m.u./kg]
KD	P1	KD_L1	16	24	270	350	30	60
KD	P2	KD_L1	16	24	380	500	45	90
KD	P1	KD_L1	40	48	270	350	30	60
KD	P2	KD_L1	40	48	380	500	45	90
KD	P1	KD_L1	112	120	810	1.050	30	60
KD	P2	KD_L1	112	120	1,140	1.500	45	90
KD	P1	KD_L1	232	240	1,350	1.750	30	60
KD	P2	KD_L1	232	240	1,900	2.500	45	90

Table 4.S2 – Process 1: characteristics of the processing units.

Unit	Min. volume	Max. volume	Unit operating Costs [m.u.]
H1	10	100	10
R1	8	80	8
R2	5	50	5
F1	20	200	20

Table 4.S3 – Process 1: materials initial, minimum and maximum availability, inventory costs and storage policy.

Resources	Init. availability [kg]	Max. availability [kg]	Inventory cost [m.u.]	Storage policy
FEEDA	10,000	10,000	0	UIS
FEEDB	10,000	10,000	0	UIS
FEEDC	10,000	10,000	0	UIS
HOTA	0	0	0.2	NIS
INTBC	0	0	0.3	NIS
INTAB	0	200	0.01	FIS
IMPE	0	0	0.04	NIS
P1	10,000	10,000	0.1	UIS
P2	10,000	10,000	0.06	UIS

Table 4.S4 – Process 2: products demand, delivery dates, price and missing delivery costs.

Network	Product	Lot product	Earliest delivery date	Latest delivery date	Minimum lot size [kg]	Maximum lot size [kg]	Price [m.u./kg]	Miss deliveries costs [m.u./kg]
KL	P1	PA_L1	40	48	10	30	50	100
KL	P2	PA_L1	40	48	20	60	60	120
KL	P3	PA_L1	40	48	30	50	30	60
KL	P4	PA_L1	40	48	5	30	20	40
KL	P5	PA_L1	40	48	10	25	45	90
KL	P1	PA_L1	112	120	30	90	50	100
KL	P2	PA_L1	112	120	60	180	60	120
KL	P3	PA_L1	112	120	90	150	30	60
KL	P4	PA_L1	112	120	15	90	20	40
KL	P5	PA_L1	112	120	30	75	45	90
KL	P1	PA_L1	232	240	50	150	50	100
KL	P2	PA_L1	232	240	100	300	60	120
KL	P3	PA_L1	232	240	150	250	30	60
KL	P4	PA_L1	232	240	25	150	20	40
KL	P5	PA_L1	232	240	50	125	45	90

Table 4.S5 – Process 2: characteristics of the processing units.

Unit	Min. volume	Max. volume	Unit operating costs [m.u.]
R1	3	10	1
R2	5	20	2
R3	4	10	1
R4	4	10	1
R5	4	10	1
R6	3	7	0.7
R7	3	7	0.7
R8	4	12	1.2
R9	4	12	1.2

Table 4.S6 – Process 2: materials initial, minimum and maximum availability, inventory costs and storage policy.

Resources	Init. availability [kg]	Max. availability [kg]	Inventory cost [m.u.]	Storage policy
STATE1	1,000	1,000	0	UIS
STATE2	10	30	0.1	FIS
STATE3	10	30	0.02	FIS
STATE4	0	15	0.02	FIS
STATE5	10	30	0.01	FIS
STATE6	0	0	0.2	NIS
STATE7	0	10	0.05	FIS
STATE8	0	10	0.05	FIS
STATE9	0	10	0.05	FIS
STATE10	0	0	0.2	NIS
STATE11	0	0	0.2	NIS
STATE12	0	10	0.01	FIS
STATE13	0	0	0.2	NIS
STATE14	0	10	0.01	FIS
P1	0	1,000	0.09	UIS
P2	0	1,000	0.09	UIS
P3	0	1,000	0.25	UIS
P4	0	1,000	0.25	UIS
P5	0	1,000	0.25	UIS

Table 4.S7 – Process 3: products demand, delivery dates, price and missing delivery costs.

Network	Product	Lot product	Earliest delivery date	Latest delivery date	Minimum lot size [kg]	Maximum lot size [kg]	Price [m.u./kg]	Miss deliveries costs [m.u./kg]
PA	PA	PA_L1	112	120	40	80	30	60
PB	PB	PB_L1	112	120	60	90	10	20
PC	PC	PC_L1	112	120	20	55	45	90
PA	PA	PA_L1	232	240	40	80	30	60
PB	PB	PB_L1	232	240	60	90	10	20
PC	PC	PC_L1	232	240	20	55	45	90

Table 4.8 – Process 3: characteristics of processing units.

Unit	Min. volume	Max. volume	Unit operating costs [m.u.]
R1	4	40	4
R2A	2	10	1
R2B	2	10	1
R3	3	30	3
R4	2	15	1.5
R5	4	40	4
R6	2	15	1.5
R7	4	50	5

Table 4.S9 – Process 3: materials initial, minimum and maximum availability, inventory costs and storage policy.

Resources	Init. availability [kg]	Max. availability [kg]	Inventory cost [m.u.]	Storage policy
PA_S1	1,000	1,000	0	UIS
PA_S2	0	0	0.2	NIS
PA_S3	0	1,000	0.01	UIS
PA_S4	0	50	0.01	FIS
PA_S5	0	0	0	ZW
PA	0	1,000	0.3	UIS
PB_S1	1,000	1,000	0	UIS
PB_S2	0	0	0	ZW
PB_S3	0	50	0.05	FIS
PB	0	1,000	0.2	UIS
PC_S1	1,000	1,000	0	UIS
PC_S2	0	0	0	ZW
PC_S3	0	100	0.04	FIS
PC_S4	0	1,000	0.02	UIS
PC_S5	0	0	0.3	NIS
PC	0	1,000	0.25	UIS

Table 4.S10 – Process 4: products demand, delivery dates, price and missing delivery costs (single lot without blending)

Network	Product	Lot product	Earliest delivery date	Latest delivery date	Minimum lot size [kg]	Maximum lot size [kg]	Price [m.u./kg]	Miss deliveries costs [m.u./kg]
PA	PA	PA_L1	16	24	3,000	5,500	15	30
PB	PB	PB_L1	16	24	2,500	7,000	20	40
PC	PC	PC_L1	16	24	4,500	6,000	50	100
PA	PA	PA_L1	40	48	3,000	5,500	15	30
PB	PB	PB_L1	40	48	2,500	7,000	20	40
PC	PC	PC_L1	40	48	4,500	6,000	50	100
PA	PA	PA_L1	112	120	9,000	16,500	15	30
PB	PB	PB_L1	112	120	7,500	21,000	20	40
PC	PC	PC_L1	112	120	13,500	18,000	50	100
PA	PA	PA_L1	232	240	15,000	27,500	15	30
PB	PB	PB_L1	232	240	12,500	35,000	20	40
PC	PC	PC_L1	232	240	22,500	30,000	50	100

Table 4.S11 – Process 4: products demand, delivery dates, price and miss deliveries costs (multiple lots with blending)

Network	Product	Lot product	Earliest delivery date	Latest delivery date	Minimum lot size [kg]	Maximum lot size [kg]	Price [m.u./kg]	Miss deliveries costs [m.u./kg]
PA	PA	PA_L1	40	48	2,000	2,500	15	30
PA	PA	PA_L2	40	48	2,000	2,500	15	30
PB	PB	PB_L1	40	48	2,000	2,500	20	40
PB	PB	PB_L2	40	48	2,000	2,500	20	40
PC	PC	PC_L1	40	48	6,000	7,000	50	100

Table 4.S12 – Process 4: characteristics of processing units.

Unit	Min. volume	Max. volume	Unit operating costs [m.u.]
U1	50	5,000	500
U2	40	4,000	400
U3	20	2,000	200
U4	30	3,000	300
U5	20	2,000	200
F1	40	4,000	400
F2	20	2,000	200
D1	30	3,000	300

Table 4.S13 – Process 4: materials initial, minimum and maximum availability, inventory costs and storage policy.

Resources	Initial availability [kg]	Max. availability [kg]	Inventory cost [m.u.]	Storage policy
PA_S1	0	0	0.2	NIS
PA_S2	0	0	0.4	NIS
PA_S3	0	0	0.1	NIS
PA	0	10,000	0.05	FIS
PB_S1	0	5,000	0.3	FIS
PB_S2	0	0	0.1	NIS
PB	0	10,000	0.03	FIS
PC_S1	0	0	0	ZW
PC_S2	0	0	0	ZW
PC_S3	0	0	0.3	NIS
PC	0	10,000	0.1	FIS

Table 4.S14 – Process 4: changeovers time between products and units.

Unit	PA	PB	PC
PA	c_r+1	c_r+2	c_r+2
PB	c_r+2	c_r+1	c_r+2
PC	c_r+2	c_r+2	c_r+1

Table 4.S15 – Process 4: changeovers time per unit.

Unit	c_r
U1	3
U2	3
U3	1
U4	2
U5	1
F1	3
F2	1
D1	3

Table 4.S16 – Process 4: costs structure.

Model/process/ horizon/grid	VP [m.u.]	SC [m.u.]	OC [m.u.]	MC [m.u.]	GC [m.u.]	Profit [m.u.]
M1.2/P4/24/1	522500	3,032	8,500	0.0	800.00	510,167.8
M2/P4/24/1	522500	3,032	8,300	0.0	0.00	511,167.8
M1.2/P4/48/1	1045000	6,807	16,800	0.0	2200.00	1,019,193.5
M2/P4/48/1	1045000	6,463	16,200	0.0	0.00	1,022,336.5
M1.2/P4/120/1	2536000	79,640	40,500	0.0	7500.00	2,408,360.0
M2/P4/120/1	2612500	71,843	41,200	0.0	0.00	2,499,456.8
M1.2/P4/240/1	4522000	352,857	73,000	200,000.0	11500.00	3,884,643.0
M2/P4/240/1	5184500	289,736	84,900	0.0	0.00	4,809,863.8

References

Barbosa-Povoa, A. P. (2007). A critical review on the design and retrofit of batch plants. Computers & Chemical Engineering, 31, 833-855.

Barbosa-Povoa, A. P., & Macchietto, S. (1994). Detailed design of multipurpose batch plants. Computers & Chemical Engineering, 18, 1013-1042.

Castro, P., Barbosa-Povoa, A. P., & Matos, H. A. (2003). Optimal periodic scheduling of batch plants using RTN-based discrete and continuous-time formulations: A case study approach. Industrial & engineering chemistry research, 42, 3346-3360.

Castro, P., Barbosa-Póvoa, A. P., Matos, H. A., & Novais, A. Q. (2004). Simple continuous-time formulation for short-term scheduling of batch and continuous processes. Industrial & engineering chemistry research, 43, 105-118.

Castro, P., Erdirik-Dogan, M., & Grossmann, I. (2008). Simultaneous batching and scheduling of single stage batch plants with parallel units. AIChE Journal, 54, 183-193.

Castro, P., Novais, Q., & Carvalho, A. (2008). Optimal equipment allocation for high plant flexibility: An industrial case study. Industrial & engineering chemistry research, 47, 2742-2761.

Castro, P. M., Harjunkoski, I., & Grossmann, I. E. (2009). New continuous-time scheduling formulation for continuous plants under variable electricity cost. Industrial & engineering chemistry research, 48, 6701-6714.

Floudas, C. A., & Lin, X. (2004). Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review. Computers & Chemical Engineering, 28, 2109-2129.

Grossmann, I. (2012). Advances in mathematical programming models for enterprise-wide optimization. Computers & Chemical Engineering.

Grossmann, I. E. (2002). Review of nonlinear mixed-integer and disjunctive programming techniques. Optimization and Engineering, 3, 227-252.

Ierapetritou, M., & Floudas, C. (1998). Effective continuous-time formulation for short-term scheduling. 1. Multipurpose batch processes. Industrial & engineering chemistry research, 37, 4341-4359.

Janak, S. L., Lin, X., & Floudas, C. A. (2004). Enhanced continuous-time unit-specific event-based formulation for short-term scheduling of multipurpose batch processes: Resource constraints and mixed storage policies. Industrial & engineering chemistry research, 43, 2516-2533.

Kallrath, J. (2002). Planning and scheduling in the process industry. OR spectrum, 24, 219-250.

Kondili, E., Pantelides, C., & Sargent, R. (1993). A general algorithm for short-term scheduling of batch operations--I. MILP formulation. Computers & Chemical Engineering, 17, 211-227.

Li, Z., & Ierapetritou, M. (2008). Process scheduling under uncertainty: Review and challenges. Computers & Chemical Engineering, 32, 715-727.

Liu, Y., & Karimi, I. (2007). Novel continuous-time formulations for scheduling multi-stage batch plants with identical parallel units. Computers & Chemical Engineering, 31, 1671-1693.

Liu, Y., & Karimi, I. (2008). Scheduling multistage batch plants with parallel units and no interstage storage. Computers & Chemical Engineering, 32, 671-693.

Maravelias, C. T., & Grossmann, I. E. (2003). New general continuous-time state-task network formulation for short-term scheduling of multipurpose batch plants. Industrial & engineering chemistry research, 42, 3056-3074.

Maravelias, C. T., & Sung, C. (2009). Integration of production planning and scheduling: Overview, challenges and opportunities. Computers & Chemical Engineering, 33, 1919-1930.

Méndez, C., Henning, G., & Cerda, J. (2001). An MILP continuous-time approach to short-term scheduling of resource-constrained multistage flowshop batch facilities. Computers & Chemical Engineering, 25, 701-711.

Méndez, C. A., & Cerdá, J. (2003). Dynamic scheduling in multiproduct batch plants. Computers & Chemical Engineering, 27, 1247-1259.

Mendez, C. A., Cerda, J., Grossmann, I. E., Harjunkoski, I., & Fahl, M. (2006). State-of-the-art review of optimization methods for short-term scheduling of batch processes. Computers & Chemical Engineering, 30, 913-946.

Moniz, S., Barbosa-Póvoa, A. P., & Pinho de Sousa, J. (2012). Regular and non-regular production scheduling of multipurpose batch plants. Proceedings of the 22nd European Symposium on Computer Aided Process Engineering.

Pantelides, C. C. (1994). Unified frameworks for optimal process planning and scheduling. In (pp. 253-274): Cache Publications New York.

Papageorgiou, L. G., & Pantelides, C. C. (1993). A hierarchical approach for campaign planning of multipurpose batch plants. Computers & Chemical Engineering, 17, S27-S32.

Pinto, J. M., & Grossmann, I. E. (1995). A continuous time mixed integer linear programming model for short term scheduling of multistage batch plants. Industrial & engineering chemistry research, 34, 3037-3051.

Pinto, T., Barbosa-Póvoa, A. P. F. D., & Novais, A. Q. (2005). Optimal design and retrofit of batch plants with a periodic mode of operation. Computers & Chemical Engineering, 29, 1293-1303.

Schilling, G., & Pantelides, C. (1996). A simple continuous-time process scheduling formulation and a novel solution algorithm. Computers & Chemical Engineering, 20, S1221-S1226.

Shah, N., Pantelides, C., & Sargent, R. (1993). A general algorithm for short-term scheduling of batch operations--II. Computational issues. Computers & Chemical Engineering, 17, 229-244.

Shaik, M. A., & Floudas, C. A. (2007). Improved unit-specific event-based continuous-time model for short-term scheduling of continuous processes: Rigorous treatment of storage requirements. Industrial & engineering chemistry research, 46, 1764-1779.

Sundaramoorthy, A., & Karimi, I. (2005). A simpler better slot-based continuous-time formulation for short-term scheduling in multipurpose batch plants. Chemical engineering science, 60, 2679-2702.

Sundaramoorthy, A., & Maravelias, C. T. (2008). Simultaneous batching and scheduling in multistage multiproduct processes. Industrial & engineering chemistry research, 47, 1546-1555.

Sundaramoorthy, A., & Maravelias, C. T. (2011a). Computational Study of Network-Based Mixed-Integer Programming Approaches for Chemical Production Scheduling. Industrial & engineering chemistry research.

Sundaramoorthy, A., & Maravelias, C. T. (2011b). A general framework for process scheduling. AIChE Journal, 57, 695-710.

Varma, V., Reklaitis, G., Blau, G., & Pekny, J. (2007). Enterprise-wide modeling & optimization--An overview of emerging research challenges and opportunities. Computers & Chemical Engineering, 31, 692-711.

Verderame, P. M., Elia, J. A., Li, J., & Floudas, C. A. (2010). Planning and Scheduling under Uncertainty: A Review Across Multiple Sectors. Industrial & engineering chemistry research, 49, 3993-4017.

Vooradi, R., & Shaik, M. A. (2012). Improved Three-Index Unit-Specific Event-Based Model for Short-Term Scheduling of Batch Plants. Computers & Chemical Engineering.

Wassick, J. M., & Ferrio, J. (2011). Extending the Resource Task Network for Industrial Applications. Computers & Chemical Engineering.

5 Paper 3: A Solution Methodology for Scheduling Problems in Batch Plants

Samuel Moniz,[a,c] Ana Paula Barbosa-Póvoa,[a] Jorge Pinho de Sousa[b,c]

[a]*Centro de Estudos de Gestão, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal*

[b]*INESC TEC, 4200-465, Porto, Portugal*

[c]*Faculdade de Engenharia da Universidade do Porto, 4200-465, Porto, Portugal*

This chapter is under review in the journal Industrial & Engineering Chemistry Research.

Abstract

This paper proposes a solution methodology for the production scheduling of batch plants. The methodology is defined by an integrated approach that simultaneously considers the representation of the scheduling problem, the optimization model and the decision-making process. A problem representation and a mixed integer linear programing (MILP) model are developed and applied to solve a real world scheduling problem from the chemical-pharmaceutical industry. The main advantage of this approach is that it includes a general process representation that can be used across several departments of the company. Moreover, we also discuss general development and implementation challenges of optimization methods for the process industry, and we provide some guidelines to mitigate existing problematic issues in this domain.

Keywords: scheduling; optimization; decision-making; enterprise-wide optimization; mixed-integer linear programming

5.1 Introduction

Decision-making in the process industry tends to be inherently complex, since it may involve strategic, tactical and operational decisions in very dynamic manufacturing systems. In particular, planning and scheduling decisions have a huge importance due to their interdependency with other functions, such as sales, procurement, production execution, and control. Hence, the integration of optimization methods to support these decisions caught the interest of many industrial companies. Model-based applications are seen as a way to improve competitiveness, to increase profitability, and also to reshape the product portfolio and to facilitate product and process innovations (Klatt & Marquardt, 2009).

In the past years, many academic and industrial efforts have been done to develop and implement model-based applications in manufacturing systems (Mendez et al., 2006). The major achievements in the area include exact, non-exact and hybrid methods, and also conceptual frameworks, ontologies and problem representations. Exact methods include Mixed Integer Linear Programming (MILP), Mixed Integer Nonlinear Programming (MINLP) and Constraint Programing (CP) models. Non-exact methods are usually based on heuristics, meta-heuristics and artificial intelligence approaches. Hybrid methods combine the previous methods so as to build more efficient approaches. On the other hand, conceptual frameworks aim at defining the scope of the different problems addressed by the Process Systems Engineering (PSE) community, and aim at proposing general integration schemes. Ontologies attempt to clarify concepts and their relations. Finally, general problem representations attempt to provide unified and unambiguous views of planning and scheduling problems.

Although these developments clearly represent a huge progress on the integration of optimization methods with the decision-making processes, there are some open issues that have recurrently been reported by the literature. The most common ones are related to the computational performance, modeling uncertainty, multiscale optimization, or the modeling task itself. The modeling challenge is surely a complex issue, since it deals with the design of models targeting their integration with the companies decision-making processes (Grossmann, 2005).

In this paper, we propose a methodology for the integration of scheduling model-based approaches with the decision-making processes. In particular, we address a scheduling problem in the chemical-pharmaceutical industry from an integrated perspective. Issues related to the problem description, modeling and implementation of scheduling models in batch plants are discussed, and considered in the proposed methodology. This work was strongly motivated from the need of solving in an integrated way and in close collaboration with a company, their day to day scheduling problems.

The rest of the paper is structured as follows. Section 5.2 presents several conceptual frameworks that have been proposed to define the decision-making levels of manufacturing systems. Section 5.3 reviews contributions from the literature addressing real world scheduling problems. Sections 5.4 and 5.5 describe the proposed methodology. We start by defining the concepts used and then we apply the solution methodology in a real world scheduling problem from the chemical-pharmaceutical industry. In section 5.6, we discuss the challenges related to the adoption of optimization methods by the industry and we present some implementation guidelines. Section 5.7 states research opportunities in the area and, finally, in section 5.8, concluding remarks are presented.

5.2 Conceptual Frameworks

Planning and scheduling are surely two critical activities performed by industrial companies. They involve the allocation of limited resources to operations that occur in given time windows. Pinedo (2002) defined scheduling as a decision-making process that deals with the allocation of resources to tasks over time, and considering one or more objectives. Stephanopoulos and Reklaitis (2011) defined planning and scheduling of process operations as a subarea of Process Systems Engineering (PSE) that deals with models, methods and tools for supporting technical decisions related to the safety, efficiency and reliability of the execution of the manufacturing functions of a process industry enterprise. These definitions are wide enough to incorporate relevant interactions with strategic areas such as sales and forecasting and with operational areas such as production execution, control and dispatching. Several authors have explored this area and have proposed conceptual frameworks (depicted in appendix) that we will briefly describe for a better understanding of the planning and scheduling functions.

Considering a logistics perspective, Meyr et al. (2005) presented the supply chain planning matrix, where planning activities are categorized in terms of time horizon and process: a) the long-term planning, called *strategic network planning* deals with the structure of the supply chain; b) the mid-term planning, is responsible for the determination of production targets, distribution of the production and capacity management; and c) the short-term planning accounts for production planning and scheduling, *i.e.*, operational decisions such as lot-sizing and tasks sequencing.

On the process operations, Bassett et al. (1996) presented a decision-making hierarchy that integrates planning, scheduling and control. The perspective supported by the authors is that model-based methodologies offer the most effective framework for integrating all these decisions. Nevertheless, due to the variety and scope of strategic and operational decisions, the authors claim that no single model would be sufficient to handle all aspects. Pinedo (2002) proposed a similar framework for generic manufacturing systems. Scheduling is positioned between production planning and shop-floor control. The decision-making process is clearly hierarchical and allows bi-directional information flows. The planning process starts with a master production planning for determining the production quantities and due dates, and to do the initial assessment of the production capacity. This data goes into the Materials Requirements Planning (MRP) that is responsible for launching orders and ensuring that the materials required for production are available. The scheduling function receives the orders from the MRP and performs the sequencing. Orders are then dispatched following the production execution. The closed-loop information flow reinforces the possibility to revise the scheduling, the MRP, or the master production planning whenever necessary, and therefore the system accuracy.

On the production execution, Harjunkoski et al. (2009) and Engell and Harjunkoski (2012) presented the *automation pyramid* for discussing the integration of planning, scheduling and control layers. The bottom level of the pyramid is composed by the *Control systems/sensors* and is mainly related to hardware/software components. The middle level is the *Manufacturing Execution System (MES)* and deals with more advanced production control algorithms, scheduling, maintenance, inventory control, quality assurance, materials and energy control. The top of the pyramid is in general based on the *Enterprise Resource Planning (ERP)* system and is concerned with the long-term strategic and tactical planning decisions, performing business-related functions such as *Available-to-Promise (ATP*) checks. According to the authors these levels are not fully standardized and their integration heavily depends on the characteristics of each company.

Standards are also being used for the definition of concepts and to perform the integration of the various subsystems of manufacturing environments. Two standards from ANSI/ISA (S88 and S95) are often referred in the literature. ANSI/ISA-88 (1995) provides models for integrating information related to the control of batch processes, and ANSI/ISA-95 (2000) has models for the integration of enterprise and control systems. The Purdue Reference Model, presented in the S95 standard, describes the main components of an enterprise system, their functionalities and interactions.

From a functional point of view, decision-making processes require infrastructures capable to effectively support information gathering, data integration and models development, as mentioned by Venkatasubramanian et al. (2006). These authors propose an *information centric* infrastructure based on an ontology to support product and process development of active-pharmaceutical ingredients (API). This approach provides a coherent knowledge base that can be used by software tools and models to promote information sharing. On the same line, Muñoz et al. (2010) developed an ontology for batch processes, considering the scheduling and the control levels in a closed loop. Although the results presented by these authors are very interesting, substantial challenges will surely arise when implementing these frameworks in industrial facilities.

Klatt and Marquardt (2009) presented an overview of methods and tools developed in the context of PSE. The authors argue that the development of user-friendly tools for industrial practitioners is still necessary. With a similar opinion as Bassett et al. (1996), Klatt and Marquardt (2009) state that emphasis should be put on model-based applications and in the development of methodologies in which the economic impact and advantages are obvious at first glance.

Although the considerable achievements done by the academia in the development of new scheduling formulations and encouraging solution approaches, the adoption of optimization planning tools by the industry is still poor (Henning, 2009). Reasons for this are related with the way the information context is considered by these tools, and are associated to an inadequate definition of the business process workflows. Stephanopoulos and Reklaitis (2011) recognized that there are important research opportunities in the development of high level but flexible representations of the scheduling problems and innovative graphical representations, which would promote the adoption of advanced planning systems.

In this paper, we develop a model-based methodology for performing scheduling in the chemical-pharmaceutical industry. We take into account the company functions that contribute for building the production schedules, and we propose a methodology that integrates the definition of the scheduling problem, the optimization model, and the decision-making process.

5.3 Scheduling in the Process Industry

In this section, we review some case-specific contributions that address the scheduling problem in batch plants. We briefly describe the models available to tackle batch scheduling problems, giving emphasis to real world applications.

5.3.1 Models for Scheduling

Significant academic achievements have been done concerning modeling and solving batch planning and scheduling problems. Some relevant recent reviews on this topic provide a rather comprehensive survey on the domain (Kallrath, 2005; Mendez et al., 2006; Barbosa-Povoa, 2007; Li & Ierapetritou, 2008; Maravelias & Sung, 2009; Verderame et al., 2010).

Mendez et al. (2006) classified scheduling problems according to the network of processing tasks. Thus, we may have sequential and network processes. In sequential processes the task-batch entity is preserved, thus batch mixing and splitting are not allowed. On the contrary, network processes have arbitrary topologies and allow batch mixing and splitting. Looking just at models suitable for network processes, we may have continuous-time formulations based on unit specific events (Ierapetritou & Floudas, 1998; Janak et al., 2004; Shaik & Floudas, 2007; Vooradi & Shaik, 2012) or based on global events (Schilling & Pantelides, 1996; Castro et al., 2001; Maravelias & Grossmann, 2003; Sundaramoorthy & Karimi, 2005). Relevant contributions have also been made in what concerns discrete-time models (Kondili et al., 1993; Shah et al., 1993; Barbosa-Povoa & Macchietto, 1994; Pantelides, 1994; Pinto et al., 2005; Sundaramoorthy & Maravelias, 2011b; Wassick & Ferrio, 2011), and on the comparison of discrete-time and continuous-time models see (Floudas & Lin, 2004; Castro & Grossmann, 2005; Sundaramoorthy & Maravelias, 2011a).

5.3.2 Real-World Scheduling Problems

Several works on real world scheduling problems and in different types of industries can be identified in the literature, where different models and solution approaches have been proposed.

For example in the pharmaceutical industry, Amaro and Barbosa-Póvoa (2008a) proposed a sequential modeling approach for the planning and scheduling of supply chains. Two MILP discrete-time formulations, for planning and scheduling problems, are developed and then linked by setting common time domain bounds. The solution approach is applied to a real pharmaceutical supply chain producing several products such as injection drugs, tablets and oral suspensions. Multistage multiproduct scheduling problems have been tackled by Kopanos et al. (2010), Stefansson et al. (2011) and Castro et al. (2009). Kopanos et al. (2010) and Castro et al. (2009) have used similar decomposition strategies to obtain solutions in reasonable computational times. Both solution approaches attempt to reduce the computational complexity of the scheduling problem by scheduling orders sequentially and improve the schedule by applying reordering procedures. Kopanos et al. (2010) proposed general precedence and unit-specific general precedence models, while Castro et al. (2009) proposed an unit-specific continuous-time formulation. Stefansson et al. (2011) compared a discrete-time formulation based on (Kondili et al., 1993; Shah et al., 1993), and a general precedence continuous-time formulation based on (Méndez et al., 2001), in a scheduling problem of a secondary pharmaceutical production system. To tackle the combinatorial complexity of the MILP models, the authors have applied a decomposition algorithm that prioritizes the scheduling of the bottleneck stage. In this way, the problem is decomposed into smaller problems that are solved separately. Results showed that the continuous-time formulation provides more accurate solutions and that it can be used to solve larger instances. Susarla and Karimi (2010) developed a unit slot continuous-time model to the campaign planning problem of the pharmaceutical industry, giving emphasis to the decision-making process. They studied several real scenarios considering different resources allocation profiles, safety stock limits, minimum campaign lengths, maintenance actions and sequence-dependent changeovers. They remark that although planning is usually performed by the planning department, it is a collaborative process that seeks data from several other departments (sales, procurement, laboratory, maintenance and higher management).

Addressing the steel making process scheduling, Harjunkoski and Grossmann (2001) developed a decomposition algorithm that relies on splitting the original problem into smaller subproblems, by exploring its special structure. The algorithm involves three MILP models and one LP model solved in a sequential solution strategy. Later, addressing also a scheduling problem of a steel plant, Harjunkoski et al. (2011) discussed the implementation issues and benefits of planning and scheduling optimization. The authors state that reusability, flexibility and configurability are relevant aspects that must be considered when encapsulating mathematical models in software applications to be used in industrial environments. Pacciarelli and Pranzo (2004) proposed an alternative graph formulation and a heuristic search strategy (beam search), and Li et al. (2012) developed a unit-specific event continuous-time formulation and present an extension of a rolling horizon algorithm.

For the production scheduling in the polymer industry, Schulz et al. (1998) formulated a discrete-time model and a continuous-time non-linear model (MINLP) for a real case of a chemical batch plant producing expandable polystyrene. Algorithms have been developed to produce solutions in reasonable time. Considering the same scheduling problem, Wang et al. (2000) have applied a genetic algorithm, and Till et al. (2007) addressed uncertainty by using a two-stage stochastic integer programming model and proposing a hybrid evolutionary algorithm to solve the stochastic problem. Castro et al. (2003) explored the optimal periodic schedule of a resource constrained industrial problem of the pulp industry, through the use of discrete-time and continuous-time Resource-Task Network (RTN) based mathematical formulations. Adequate solution strategies were proposed for both formulations. While the exact optimal solution to the problem was achieved using the discrete-time formulation, the same was not true for the continuous-time formulation.

In the scheduling of chemical batch plants, Erdirik-Dogan et al. (2008) formulated a MILP model for the short-term scheduling of parallel batch reactors. They concluded that for addressing mid and long term scheduling horizons specialized solution algorithms must be developed. Erdirik-Dogan and Grossmann (2008) developed a time slot continuous-time model and a bi-level decomposition algorithm that involves solving iteratively an aggregated model and a detailed scheduling model.

Considering *Enterprise-Wide Optimization* (EWO) in complex production systems, Wassick (2009) presented the case of Dow Chemical Company, and discussed opportunities for the integration of design, planning and scheduling optimization models in the industry. The problem of waste disposal scheduling is solved using the RTN discrete-time formulation. Moreover, the author presented some useful considerations on modeling and implementation. So, to the company, the choice of the discrete-time RTN relied on the simplicity and generality of the formulation, due to an uniform treatment of all resources. A creative definition of the production resources allows solving a variety of scheduling problems without changing the model (and the code). Simple linear representations of the processes are adequate for long time frames or more strategic decisions, but for short time frames or operational decisions, it becomes necessary to account for the non-linearities of the chemical processes. For this author, the greatest modeling challenge concerns capturing complex operating policies. In these cases, it is recommended to negotiate simplifications with the decision makers, instead of dealing with complicated constraints. Moreover, the process design, planning and scheduling integration, and the representation of uncertainty and risk, should be viewed as critical. An interesting case-specific aspect about the implementation is that during the first year of operation, the scheduling model was used together with the existing scheduling procedure, in order to compare both methods and to gain confidence in the model.

In general, Applequist et al. (1997) pointed out four practical issues that make planning and scheduling problems particularly difficult to address, namely: a) *social considerations* –manufacturing is considered a cooperative activity in the company, and the "planning and scheduling" function is viewed as having the responsibility to orchestrate this cooperation; b) *dynamic nature* – the active environment of the manufacturing system requires flexible and scalable planning and scheduling tools that must be able to adapt to different production scenarios; c) *information intensity* – even relatively small planning and scheduling problems require a considerable amount of data, and this creates additional complications concerning data management; and d) intrinsic *combinatorial character* – leading to significant mathematical challenges to solve these problems.

The relevance of these issues, and the practical need to address them, are the main motivation for this work. Our research is in fact pursued with the aim of designing new optimization based decision-support tools, for solving planning and scheduling problems in process batch plants.

5.3.3 Integration of Optimization Models in Industry

As referred it is clear that there is still a lot of work to do concerning the integration of optimization models in the decision-making processes of the companies. The literature addressing models and solution approaches for planning and scheduling problems is mainly focused on time performance and comparison between models. When addressing the quality of the solutions, few confront the solutions obtained by the models with the solutions obtained by the planners, and just a small number of works address practical issues associated to modeling and implementation of optimization methods in industrial companies. Models are rarely evaluated in the context where the information is available, thus they simply do not consider the internal decision-making processes of the company. This interaction is missing and it would surely provide valuable information for improvement of the optimization methods.

Although we also consider that, in practical terms, time to obtain solutions is the most critical issue of many models solving scheduling problems, research should also focus in other aspects that are determinant to integrate optimization methods in industrial practices. The development of systematic approaches for structuring the scheduling problems and the inclusion of the decision-making processes into models that can be rapidly understood by the industrial practitioners, are essential issues that have been somehow neglected. These issues are being addressed in this paper, where we aim at contributing to reduce the existing gap between the design and development of scheduling models and their applicability to real industry problems.

5.4 Methodology Conceptual Framework

The conceptual framework, developed in this work, proposes a generic and systematic approach for tackling scheduling problems in real production systems. It attempts to identify the key issues for the definition of a scheduling problem and for the integration of optimization models in industrial companies. In this section, we discuss the components of the methodology and then, in section 5.5, we present the solution of a real world problem from the chemical-pharmaceutical industry to demonstrate its applicability.

5.4.1 Key Components

The methodology is defined by three main components, as shown in Figure 5.1. The *Problem Representation* component that is related to the interpretation of the scheduling problem, and it is used as an interface with the decision-makers and to capture data for the optimization model. The *Optimization* component that deals with case-specific models and solution approaches developed to solve the scheduling problem. The *Decision-making* component that has to do with the analysis of the solution pool provided by the optimization component, and involves the visualization and user-interface interactivity required to support the decision-making process.

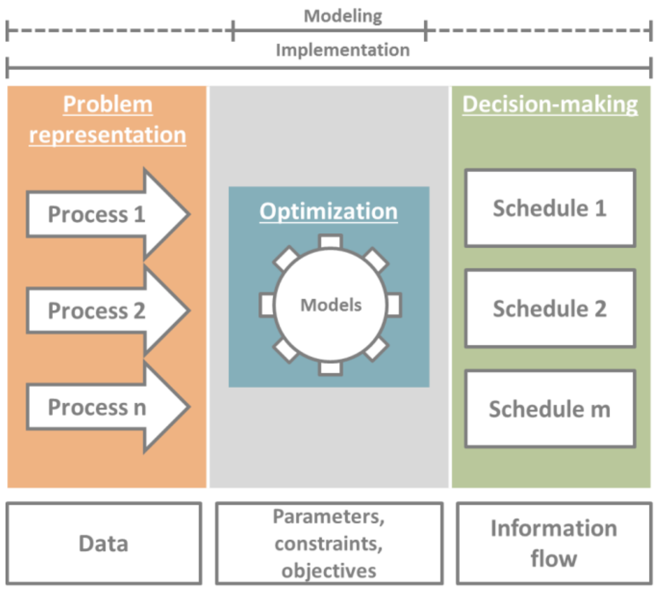


Figure 5.1 – Conceptual framework for the solution methodology.

In this context, we consider two main sets of tasks: *Implementation* and *Modeling*. Implementation includes all tasks required to place the application running in the company and it is typically assigned to IT consultants. Its scope must be wide enough to incorporate all components that in the end will constitute the decision-making tool. Modeling concerns the analysis and definition of the model and is usually a task performed by academics and researchers. The scope of modeling is in many situations limited to the development and test of models and disregards the context where the data is created and gathered, and mainly how decisions are made. In our view, when addressing real-world optimization problems, the scope of the modeling task must be broadened to include the data context and to encompass the information flow of the decision-making processes. In this paper, we explore the idea that the modeling task must be extended to define more complex interactions between the representation of the problem and the decision-making process (see Figure 5.1). This will surely ask for a deeper collaboration between academics, industrial practitioners and IT consultants.

Note that although the focus of this methodology is on scheduling problems, we think that it can also be applied to other types of problems.

The components and their interactions will be discussed in detail in the following subsections.

5.4.2 Representation of the Scheduling Problem

As Bassett et al. (1996), we also view scheduling as an integration activity. Accordingly, scheduling problems should be represented so that different types of knowledge can be captured in a coherent way. For that, different scheduling views may be necessary in order to ensure a comprehensive representation of the problem. Such representation can have then several layers (or views) and must be able to be integrated with any model or solution approach. Grossmann et al. (1999) argue that the application of mathematical programming approaches to process design and synthesis problems require the development of superstructures for the representation of the alternatives, regardless of the detail of the model. We can say that this reasoning is also valid to planning or scheduling problems, since these problems use similar superstructures.

Although there is a general consensus that models / solution approaches must be adapted to the specific features of each case, we believe that it is possible to develop general representations of the scheduling problems that could be used later by different models. A good example of this is the State-Task Network (STN) of Kondili et al. (1993) and the RTN of Pantelides (1994) representations that are being applied to represent a variety of scheduling problems and are used by many different formulations. For example, extensions of STN to account for design and operational decisions were developed by Barbosa-Povoa and Macchietto (1994) and Amaro and Barbosa-Póvoa (2008b) developed the chain-STN to solve supply chain problems. In this way, scheduling representations could evolve independently from the model formulations and provide a coherent representation of the problems. This research direction has been recently followed by Maravelias (2012). The author proposes a framework for the description of scheduling problems in chemical industries based on the characteristics of the processes.

Furthermore, the representation of the scheduling problem is typically based on the process structure, in which the level of abstraction is a critical issue. High detail representations of the processes may allow the development of more detailed models and reach theoretically optimal solutions, but may result into models that are computationally intractable. In practice, this approach requires the involvement of industrial and academic specialists and the integration of different types of knowledge, which will easily turn into a very time consuming task. On the other hand, less detailed processes result into more simple models that are easier to solve and to manage, but may result into infeasible schedules. In summary, a careful exploitation of the problem structure is required in order to keep the balance between these tradeoffs, and here a close collaboration between academics and industrial planners must exist.

In the PSE community, planning and scheduling problems appear closely connected to process development and design problems (Barbosa-Povoa, 2007). The process design focus is to define the characteristics of products, the production tasks and the specifications of processing units. The planning and scheduling problem take often the design into account and seek the effective use of the enterprise resources (Stephanopoulos & Reklaitis, 2011). In this work, we have developed a comprehensive representation of the scheduling problem that captures the characteristics of the processes and available equipment, defining superstructures with possible production alternatives.

The representation of the scheduling problem is then a key part of the scheduling methodology that integrates with the optimization and the decision-making components.

5.4.3 Optimization

A single model would hardly be sufficient to address all types of planning and scheduling decisions. Thus, if a model-based approach is followed, then the methodology must be able to include several models, in which the links between those models play a crucial role. Furthermore, since the decision-making processes vary from company to company, case-specific models may also be developed.

Concerning the computational complexity, many scheduling models solving real world problems are considered too large to be solved to optimality in affordable time. This is due to the combinatorial nature of the problems, associated to binary decisions such as task-unit assignments, tasks sequencing, changeovers and storage tasks. Problems with a significant number of tasks and processing units and considering long scheduling horizons tend to be difficult to solve with exact methods. In these cases, alternative solution approaches can be applied to obtain satisfactory solutions in reasonable time. A discussion on scheduling models and solution approaches has been presented in section 5.3.

In summary, models and solution approaches should be built taking into account the characteristics of the problems and the decision-making processes of the companies, thus defining concise methodologies that integrate mathematical approaches with existing decision-making procedures and result in solvable models that represent adequately the reality.

5.4.4 Decision-Making

The planning information flow presented by Pinedo (2002) and the Purdue Reference Model are two comprehensive frameworks where the complexity of the planning and scheduling activities are evident. To address this complexity, the development and implementation of decision-support tools should start by addressing the core decision-making processes of the company. The approach may vary from company to company, but should always involve academics and industrial practitioners.

The scheduling process must be supported by adequate information flows between those participating in the decision-making process. Assuming that scheduling is performed collaboratively, the scheduling methodology (*Problem representation*, *Optimization*, *Decision-making*) should ensure that the necessary data is available and up to date, before being used by the optimization model. In this way, the methodology must be integrated transversally in the company, since it is common that several functions in the company can use that information and benefit from it.

The scheduling methodology presented in this paper proposes a development and implementation scheme for decision-support tools to tackle scheduling problems in the chemical-pharmaceutical industry. To clearly explain this methodology the application to a case-study is detailed below.

5.5 Scheduling Methodology – Application

The proposed conceptual framework is now applied to a real case-study from the chemical-pharmaceutical industry. The main goal of this exercise is to demonstrate how the components (*Representation of the scheduling problem, Optimization,* and *Decision-making*) can be designed in order to implement a decision-support methodology for the scheduling problems of a batch plant. For that purpose, we briefly describe the context of scheduling problem in the chemical-pharmaceutical industry and a typical scheduling decision-making process. We then present the methodology that was implemented in our case study and discuss the main decisions involved in that process. The optimization model and results are also presented.

5.5.1 The Scheduling Problem in Chemical-Pharmaceutical Industry

The chemical-pharmaceutical industry is responsible for the development and manufacturing of fine chemicals called Active Pharmaceutical Ingredients (API). Manufacturing such products involves complex and long processes that are executed under a close supervision of the regulatory authorities, with responsiveness of the manufacturing system and cost reduction being two critical aspects.

Production may simultaneously include products that are under development and products that are in commercialization, and the plant resources may be shared between these products. Products are associated to *recipes* that describe in detail the production processes. Recipes are defined by a network of production tasks that must be executed to manufacture a product (Reklaitis, 1995), and include process data such as materials consumption and production proportions, tasks processing times and characteristics of the units required by each task. Recipes differ from product to product, *i.e.*, tasks sequencing and material flows are product specific. To manufacture a single product several days of effective production time may be required, with tasks processing times varying between one hour and two days.

Cleaning of processing units, pipelines and other resources is needed to avoid cross contamination of the products. Therefore changeovers between lots of the same product and between lots of different products are present and may impose significant downtime periods.

Often the chemical-pharmaceutical industry relies on general purpose batch plants with multipurpose processing units between which connections are usually not fixed. Instead units are organized in such a way that almost all connectivity options are possible. Operations flexibility is achieved through multipurpose units, capable of executing a variety of chemical tasks, as well as, through the connections between units, that can be changed when there is a change from a product to another product. I.e., connections between units can change with the production demand.

In such plants, the most common units are reactors, filters and dryers of different volumes, packaging rooms, and auxiliary equipment like condensers, temperature systems, cleaning in place (CIP), vacuum pumps, etc. that may be attached to the processing units, thus changing their configuration with additional characteristics important for the task-unit assignment. For instance, a reactor is defined by its maximum and minimum volume, type (glass lined or stainless steel) and also by the agitation system, the temperature system, CIP, etc. The material flows are established through a complex system of pipelines and mobile vessels. Furthermore, people are critical resources and are usually considered in the medium and short-term scheduling. This happens because tasks require specialized manpower to execute or control the production.

The planning and scheduling functions are typically a responsibility of the *planning* department of the company. However other areas, such as sales, procurement and R&D, are also involved. These areas contribute with data inputs relevant for planning and scheduling, and may as well do the analysis of the solutions.

Planning is in general done for a time horizon of up to twelve months, and involves the determination of the production quantities and a preliminary allocation of the processing units to products. But because production capacity is defined at the level of the processing units, planning is referred here as medium-term scheduling. The medium-term scheduling tends to be stable, at least for the next months, and is revised every month or whenever an unexpected event that has impact on the plan appears.

There is also the short-term scheduling that has a time horizon of up to two weeks and is revised on a daily basis. Data from the medium-term term scheduling is used as a reference for building the short-term scheduling, namely in what concerns: recipes, inventory and products demand. Decisions at this level refer to the assignment of tasks to units, and to the determination of the exact time when tasks are going to be executed. Industrial planners are therefore challenged to obtain the “best” set of processing units to manufacture each product and to obtain an effective tasks sequencing, taking into account objectives related to cost and total production time. Since recipes may have a large number of tasks and tasks may be processed through multiple units of different capacities, where sequence dependent changeovers must be respected, scheduling decisions become extremely complex.

The status of the tasks execution is continuously checked and potential delays are evaluated. Products that are under development add more complexity to the scheduling function, since they regularly impose the revision of the schedule. Schedule deviations that do not have any further impact in the plan are promptly solved, while significant delays trigger the revision of the medium-term schedule.

Although we assume here that medium and short-term scheduling are two distinct problems, they are linked because they constrain each other. Both scheduling problems should therefore use a unique problem representation, defined by the methodology proposed in this work (see section 5.5.3). This will ensure that with respect to data, both problems use the same structures, although the detail level of the models can be different. In this paper, our focus is on the short-term scheduling problem, as defined in detail in the next section.

5.5.2 Problem Statement

The problem consists then in finding the optimal scheduling of multipurpose batch plants, in which products have arbitrary network structures.

Given:

a) the recipes of the products (materials flows and proportions, tasks processing times and characteristics);
b) the processing units (including all characteristics that define the task-unit suitability);
c) the resources availability (intermediaries, final products and processing units for every time interval);
d) the demand (quantities and delivery due dates);
e) the minimum and maximum allowed lot sizes;
f) the scheduling time horizon;
g) the costs (storage, changeover, missed deliveries);
h) the economic value of the products.

Our goal is to obtain optimal production schedules by determining:

a) the task-unit suitability for a given process;
b) the task-unit assignment of the production schedules;
c) the lot sizes and product deliveries;
d) the materials inventory levels;
e) the optimal process schedule.

In this context, we have used profit maximization and cost minimization objectives, defining a short-term scheduling problem that can be solved by a linear model, with deterministic data.

5.5.3 Proposed Scheduling Methodology

The conceptual framework depicted in Figure 5.1 gave origin to the scheduling methodology shown in Figure 5.2. The three components of the methodology (*Problem representation*, *Optimization*, and *Decision-making*) are now framed by the associated activities (recipe design and cost modeling, scheduling/rescheduling, and decision-making) that interact and provide/receive data to/from the optimization model. These activities were identified in our case study as being core activities that have a huge relevance in the scheduling process. Overall, this methodology presents a detailed framework for the integration of optimization models with the scheduling decision-making processes. In the following subsections each component will be addressed in detail.

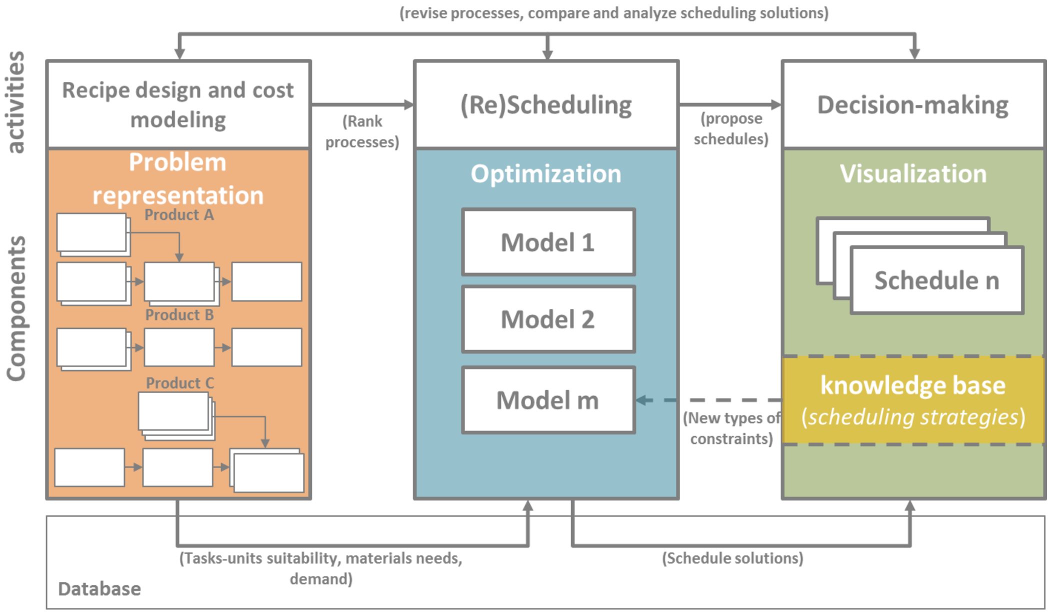


Figure 5.2 – Proposed scheduling methodology.

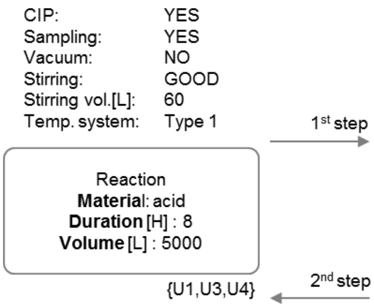
Note that this methodology views the scheduling activity as an interactive and collaborative process that may involve several departments of a company. Thus, the involved departments may provide data to the process, and revise and analyze scheduling solutions.

In order to test and validate the proposed methodology, we have, during one year, performed meetings in a regular basis with process engineers and planners. Insights from industrial practitioners revealed to be very useful in redefining the components of the methodology and the integration requirements between those components.

5.5.3.1 Representation of the Scheduling Problem

This component aims at providing a standard representation of the processes in such a way that they can be readily understood by all the participants in the scheduling problem. Having this goal in mind, we propose a novel representation of the processes in which emphasis is given to the definition of the task characteristics that will determine the task-unit assignment.

In a way similar to the STN, we design recipes through a network of *tasks* and *states*. As shown in Figure 5.3, tasks (represented as rectangles) are defined through an object that considers all the characteristics (*e.g.*, volume, task duration, need for CIP, sampling, vacuum pump, etc.) that are relevant to the determination of the suitable processing units. For example in Figure 5.3, if the task has an acid material, only units U1, U3 and U4 can be used. But if CIP and *sampling systems* are also required, only U1 can be used. The states define the materials and are represented by circles.



Charact.\Unit	U1	U2	U3	U4	U5
Type	acid(res)	basic(res)	all	acid(res)	basic(res)
CIP	•	•			•
Sampling	•		•		
Vacuum		•		•	•
Stirring vol.	50	20	100	40	110
Max vol.	10000	3000	6000	5000	2000
Temp. control	Type 1	Type 2	Type 2	Type 1	Type 3
Cost	1000	350	1750	900	1100
...					

Figure 5.3 – Mapping between tasks and processing units characteristics.

Processes are represented by the definition of tasks, material states and material flows using directed arrows, in a prototype developed in Microsoft Visio. Predefined objects are available to support the process design. So, in a first step the description of the process is done taking into account just the characteristics of the tasks, and in a second step these characteristics are automatically mapped into the characteristics of the existent processing units for the determination of the suitable units. More advanced rules can be used to account for approximate (roughly defined) characteristics such as the need for very good, good and normal stirring. Additionally, the cost modeling of the processes can also be performed, taking into account the resources involved. This can, for example, be used for ranking the alternative processes based on their cost.

Having defined the tasks and associated tasks to units, the global process representation is obtained (see Figure 5.4). The recipe design tool shown in Figure 5.4 is a prototype developed in Microsoft Visio that allows an immediate assessment of the process concerning the determination of the units capable to manufacture it. In the left hand side of the screen, it is shown the library of standard objects, in the middle we can see the process (in this case, we have a process with 6 tasks) and in the right the characteristics of the selected task are presented. The processing units suitable for each task are depicted just below the rectangle and were automatically determined through the task-unit matching characteristics as explained above.

All the data of this process is saved in a database that can be used later by any optimization system. Thus, the problem representation and optimization components are indeed independent.

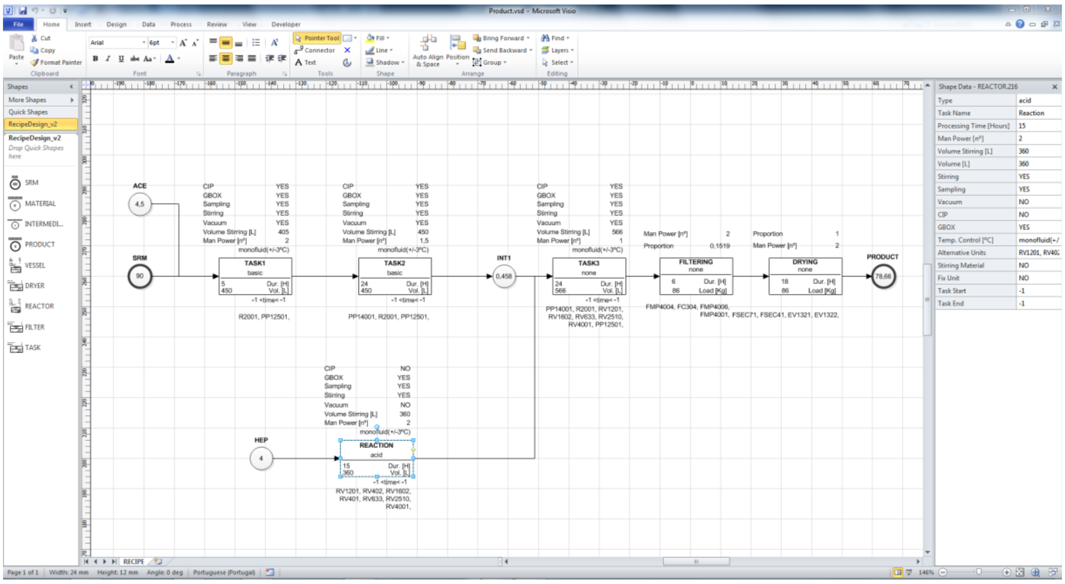


Figure 5.4 – Example of a process representation using the recipe design tool.

The main challenges in this step are related to the data management, since many data inputs are typically required to describe a single process, and to produce a representation of processes that can be used by all participants. The tests performed with the company demonstrated that the developed interface (as shown in Figure 5.4) is user friendly and can be used by all involved departments, and that the design of the processes is quite fast. These characteristics are fundamental for the planners since they are determinant to an effective usage of the tool.

5.5.3.2 Optimization

After having the processes represented in the recipe design tool, data structures are automatically generated and can be used by the scheduling model. With this approach, recipe design decisions can promptly be revised and integrated with the scheduling model.

The optimization stage is related to the *scheduling* function, but in real manufacturing systems this function is mainly used for *rescheduling*. In fact, when this function is performed, either processing units are executing tasks or planned orders have already been allocated for the near future. Then when new orders arrive, planners may have to revise the current schedule, compare and analyze scheduling solutions, or even evaluate alternative processes. Scheduling responsiveness is ensured here by an immediate assessment of the alternative processes and by the integration with the optimization model.

The solutions delivered by the model can be quantified not only by the value of the objective function, but also by the computation of several Key Performance Indicators (KPIs) done in the post-processing phase of CPLEX. For example, we may have KPIs for measuring the free capacity and volume usage of the reactors, or the missing deliveries of a scheduling solution.

The scheduling model used in the developed framework is based on the formulation proposed by Moniz et al. (2013). This is a MILP model where *time* is uniformly discretized along the scheduling horizon of interest. One particular characteristic of this model is that the material balance constraints consider explicitly the inventory carried out by each task and production lots. Production lots refer to the amount of stable intermediary or final product manufactured through a known set of tasks, units and materials, so as to keep record of lots blending operations, thus ensuring lots traceability. By following this approach, new types of constraints for modeling temporary storage in the processing units and sequence-dependent changeovers can be derived.

The indices, sets, parameters and decision variables used by the formulation are fully described in appendix.

Model

The scheduling problem considers a scheduling horizon with length T, divided into time intervals $t \in H$ of equal and fixed duration. Scheduling decisions are made through the

N_{klt} task-unit assignment/sequencing binary variables that are equal to 1 if task *k* of lot *l* starts at time interval *t*; Y_l lot production binary variables, equal to 1 if *l* lot is produced; ξ_{klt} task batch size continuous variables that determine the batch size of task *k* of lot *l* at time interval *t*; β_l lot size continuous variables that state the amount of product manufactured in lot *l*; the R^p_{krlt} and R^c_{krlt} continuous variables that define the materials *r* production *(p)* and consumption *(c)* for each task *k* of lot *l* and time interval *t*; R_{krlt} continuous variables that give the resultant materials *r* availability; Π_{krlt} continuous variables for product *r* deliveries, given by task *k*, lot *l* and time interval *t*; and B_{rt} continuous variables of backlogged demand, determined for product *r* and time interval *t*.

The model is defined by constraints (5.1) that express the fact that either processing units are allocated to production tasks or to storage operations. In other words, constrains (5.1) define the task-unit assignment and sequencing, and the temporary storage in the processing units. In the chemical industry, it is common to find processes in which material storage may occur in the processing unit where the material was produced. In these cases, units work temporarily as storage vessels until all material is consumed by subsequent tasks of the process. The first term of the constraints does the task-unit assignment and sequencing, while the second term indicates if the processing unit is performing storage ($R_{krlt} \geq 0$) for the intermediaries produced by task *k* and subject to *Non-Intermediate Storage* policy (NIS) I^{NIS}_k.

Constraints (5.2) determine the amount of resource *r* (intermediaries and final products) produced, and constraints (5.3) give the amount of resource *r* consumed (intermediaries) by task *k* of lot *l* at each time interval *t*. Parameters $\nu^p_{kr\theta}$ and $\nu^c_{kr\theta}$ give the materials production and consumption proportions of the batch size of task *k* for resource *r*. Constraints (5.4) express the materials balance for each resource *r* (intermediary or final product), task *k* and lot *l*. The amount of resource *r* available R_{krlt} in each task *k* of lot *l* is equal to the amount stored in the previous time interval, plus the amount produced R^p_{krlt}, minus the amount consumed R^c_{krlt}, plus the amount that is delivered Π_{krlt}. Note that Π_{krlt} take negative values for product deliveries and that we assume no receipts of materials occur during the scheduling horizon. Constraints(5.5) bound the resource *r* availability to a maximum value given by parameter R^{max}_{rt} and are only defined for intermediaries subject to *Finite Intermediate Storage* (FIS), *Zero-Wait*

(ZW) and *Unlimited Intermediate Storage* (UIS) policies. Concerning the materials temporarily held on the processing units (NIS), the amount of material that can be stored is bounded by the maximum capacity of the unit. Constraints (5.6) ensure that the tasks batch size ξ_{klt} is within the minimum $V_{krl}^{min}$and maximum V_{krl}^{max} capacities of resource *r* (processing unit) for task k and lot *l*. Constraints (5.7) impose that the total amount of product manufactured must be equal to the lot size β_l, and constraints (5.8) bound the lot size β_l between the minimum q_l^{min} and maximum q_l^{max} allowed size for lot *l*. Constraints (5.9) define that lot l can only be produced if lot $l-1$ has been produced.

Backlogged demand B_{rt} is defined by expressions (5.10), where B_{rt} will take a value greater than zero whenever a product delivery D_{rt} is not fulfilled, partially or totally.

Sequence-dependent changeovers are required whenever cleaning and units setup operations need to be performed, when changing the production to a new product or lot. Thus, constraints (5.11) state that if task *k* of lot *l* occurs at time interval *t*, then the first term is equal to one, and the second is forced to be zero for all tasks $k'$of lot l' and time intervals corresponding to $t - \tau_{k'} - \theta$.

Constraints (5.12) define that tasks must finish in the time horizon of interest. Constraints (5.13) impose that delivery variables Π_{krlt} cannot take values either for the time intervals outside the delivery time windows, or for resources other than final products. And expressions (5.14) define the variables domain.

$$\sum_{k\in K_r}\sum_{l\in L_r}\sum_{t'=t-\tau_k+1}^{t} N_{klt'} + \sum_{k\in K_r^p}\sum_{r'\in I_k^{NIS}}\sum_{l\in L_r}\left(\frac{R_{kr'lt}}{V_{kr'l}^{max}}\right) \leq 1 \quad \forall r \in E, t \in H \tag{5.1}$$

$$R_{krlt}^{p} = \sum_{\theta=0}^{\tau_k}\left(\nu_{kr\theta}^{p}\xi_{kl,t-\theta}\right) \quad \forall r \in I \cup P, k \in K_r^p, l \in L_r, t \in H \tag{5.2}$$

$$\sum_{k\in K_r^p} R_{krlt}^{c} = \sum_{k\in K_r^c}\sum_{\theta=0}^{\tau_k}\left(\nu_{kr\theta}^{c}\xi_{kl,t-\theta}\right) \quad \forall r \in I, l \in L_r, t \in H \tag{5.3}$$

$$R_{krlt} = \left(R_{krl}^{init}|_{t=0}, R_{krl,t-1}|_{t>0}\right) + R_{krlt}^{p} - R_{krlt}^{c} + \Pi_{krlt} \quad \forall r \in I \cup P, k \in K_r^p, l \in L_r, t \in H \tag{5.4}$$

$$0 \leq \sum_{k\in K_r^p}\sum_{l\in L_r} R_{krlt} \leq R_{rt}^{max} \quad \forall r \in I \backslash I^{NIS} \cup P, t \in H \tag{5.5}$$

$$V_{krl}^{min} N_{klt} \leq \xi_{klt} \leq V_{krl}^{max} N_{klt} \quad \forall\, r \in E, k \in K_r, l \in L_k, t \in H \tag{5.6}$$

$$\sum_{k\in K_r^p}\sum_{t\in H} R_{krlt}^{p} = \beta_l \quad \forall r \in P, l \in L_r \tag{5.7}$$

$$Y_l q_l^{min} \leq \beta_l \leq q_l^{max} Y_l \quad \forall\, l \in L \tag{5.8}$$

$$Y_{l-1} \geq Y_l \quad \forall\, l \in L: l > 1 \tag{5.9}$$

$$B_{rt} = \left(B_{r,t-1}|_{t>0}\right) + D_{rt} + \sum_{l\in L_r}\sum_{k\in K_r^p} \Pi_{\mathrm{krlt}} \quad \forall r \in P, t \in H \tag{5.10}$$

$$\sum_{k\in f_l^r} N_{klt} + \sum_{k'\in f_{l'}^r} N_{k'l't-\tau_{k'}-\theta} \leq 1 \quad \forall r \in E, l, l' \in L_r, \theta = 0, .., c_{rl'l} - 1, t \in H \tag{5.11}$$

$$\sum_{t=T-\tau_k+1}^{T} N_{klt} = 0 \quad \forall k \in K, l \in L_k \tag{5.12}$$

$$\begin{gathered} \Pi_{krlt} = 0 \quad \forall r \in P, l \in L_r, d \in D_r, k \in K_r^p, t \in H \backslash DW_{rd} \\ \Pi_{krlt} = 0 \quad \forall r \in I, L \in L_r, k \in K_r^p, t \in H \\ DW_{rd} = \left\{ t \mid r \in P, d \in D_r, t \in H :\ T_{rd}^{dd} \geq t \geq T_{rd}^{ed} \right\} \end{gathered} \tag{5.13}$$

$$\begin{gathered} R_{krlt}^{p}, R_{krlt}^{c}, R_{krlt} \in \mathbb{R}_+ \quad \forall r \in I \cup P, k \in K_r^p, l \in L_r, t \in H \\ \xi_{klt} \in \mathbb{R}_+ \quad \forall r \in E, k \in K_r, l \in L_k, t \in H \end{gathered} \tag{5.14}$$

$$\Pi_{krlt} \in \mathbb{R}_- \quad \forall r \in I \cup P, l \in L_r, k \in K_r^p, t \in H$$

$$Y_l \in \{0,1\} \quad \forall\, l \in L$$

$$N_{klt} \in \{0,1\} \quad \forall\, k \in K, l \in L_k, t \in H$$

The objective functions considered in this work are: the minimization of cost (see expression (5.15)), that involves the storage, operational, backlog and lot costs; and the maximization of profit, (see expression (5.16)), reflecting the economic value of the products.

Storage costs are associated to holding costs of intermediaries and products during the scheduling horizon (5.16a). Operational costs are related to the assignment of processing units to tasks are defined by expression (5.16b). Backlogged demand costs are given by expression (5.16c) and lot fix and variable costs are given by expression (5.16d). The economic value of the products is given by expression (5.16e).

$$\begin{aligned}\min Z1 = {} & storage\ costs(SC) + operational\ costs\ (OC) \\ & + backlog\ costs(BC) + lot\ costs\ (LC)\end{aligned} \tag{5.15}$$

$$\begin{aligned}\max Z2 = {} & value\ of\ the\ products\ (VP) - storage\ costs(SC) \\ & - operational\ costs\ (OC) - backlog\ costs(BC) \\ & - lot\ costs\ (LC)\end{aligned} \tag{5.16}$$

$$SC = \sum_{r \in I \cup P} \sum_{l \in L_r} \sum_{k \in K_r} \sum_{t \in H} (c_r^{sto} R_{krlt}) \tag{5.16a}$$

$$OC = \sum_{k \in K_r} \sum_{l \in L_r} \sum_{t \in H} \left(c_k^{op} N_{klt}\right) \tag{5.16b}$$

$$BC = \sum_{r \in P} \sum_{t \in H} c_r^{bk} B_{rt} \tag{5.16c}$$

$$LC = \sum_{l \in L} (c_l^{fix} Y_l + c_l^{var} \beta_l) \tag{5.16d}$$

$$VP = \sum_{r \in P} \sum_{l \in L_r} \sum_{k \in K_r^p} \sum_{t \in H} (-v_r \Pi_{krlt}) \tag{5.16e}$$

5.5.3.3 Decision-Making

The scheduling solutions obtained through this model are then evaluated by experienced planners and other participants involved in the scheduling process. In practice, it is desirable to produce more than one schedule, even for the same objective function, as the model may not represent the real problem due to simplifications considered. In some situations model constraints are linear approximations or aggregations designed to keep the problem computationally tractable. Thus, it could happen that solutions might not be preferred by planners or could be considered operationally infeasible. The assessment of the model will be then made by the quality of the solutions delivered and the computational time required to produce them.

To avoid this problem, several scheduling solutions are generated and compared during the decision-making process. For example, multiple scheduling solutions can be obtained by using the CPLEX solution pool feature or simply by setting different values of CPLEX stopping criteria, such as the *integrality gap* or the *time limit.* Note that these solution strategies do not guarantee that the solutions are optimal.

Although the most common constraints of the scheduling problem are known and well described in the literature, new types of constraints are often necessary when trying to solve real world scheduling problems. To address this issue, a knowledge base is kept with the purpose of describing new scheduling rules that are empirically followed by the planners. The scheduling rules are then evaluated and, if applicable, are converted into model constraints. Thus, the model is composed by a set of constraints that can be activated or deactivated in order to convey to the preferences of the planner. For example, case-specific extensions for dealing with layout, manpower and maintenance constraints may be considered in the model.

Finally, the user interface for the scheduling solutions plays also an important role, since the dynamic nature of the scheduling process requires the visualization of a considerable amount of information, as well as advanced interactive options. In our case, a prototype of a Gantt chart was built in Microsoft Excel to allow the test and assessment of the scheduling solutions. The evaluation of the produced pool of schedules is then supported by Gantt charts and additional indicators (see below) allowing the planners to choose the most adequate schedule for real implementation.

5.5.4 Case-Study Description and Results

As mentioned above, the case-study under analysis concerns the scheduling problem in a chemical-pharmaceutical industry where different products are to be produced. The decision-making tool is used to obtain optimized production plans and to perform an evaluation of alternative production processes. The scheduling decisions involve the determination of the processing units to be used by each process, the lot sizes, the total amount produced and the delivery dates. Here, recipe design decisions (as described in section 5.5.3.1) are integrated with scheduling decisions in order to predict the impact of the process in the shop-floor, guaranteeing that units are used in an efficient way. For example, the selection of the processing units to execute a given process can be done taking into account the scalability, completion time, number of processing units used, material flows, costs, etc. of the process.

Since a discrete time model is being used, a critical modeling decision concerns the definition of the length for the time intervals. A thin discretization of time would theoretically result in better solutions, but may conduct to models that are very difficult to solve. In our case, we assume a time interval (grid) of 8 hours, since tasks processing times can be roughly approximated by multiples of 8. Moreover, schedules having time intervals of 8 hours (1 working shift) work well in practice. Computational tests and discussions with the planners have shown that the computational time required to solve the short-term scheduling model is quite reasonable and acceptable in practice. The model was implemented using ILOG/CPLEX version 12.5.1, running on an Intel Xeon X5680 at 3.33GHz with 24 GB of RAM.

In the following two subsections, we present some results that show the utilization of our scheduling model (defined in section 5.5.3.2). Initially, we perform an analysis of the processes involved and we discuss tradeoffs related to the determination of the lot size, which are important to the scale-up strategies followed by the chemical-pharmaceutical industry. Then, we derive short-term production schedules for time horizons of 1 and 2 weeks.

The network studied in this work considers four processes, responsible for the production of four products (P1 to P4) and that may share 9 processing units (7 reactors of different characteristics and 2 filters-dryers). The total number of tasks is 40, with processing times varying between 6 and 64 hours. Processes data is available in the supporting information file

5.5.4.1 Evaluation of Alternative Processes

The impact of the lots definition on the cost is the first analysis to be performed. This is a relevant indicator concerning the scale-up of the lot size. Figure 5.5 depicts the minimum cost of manufacturing 30, 40, 60, 80 and 100 kg of product P1, considering different lot sizes (*e.g.* 3 lots of 10 kg = 3L10).

Table 5.1 shows some numerical values used in this analysis of the process alternatives. These results were obtained by running the scheduling model with the cost minimization objective function ($\min Z1$), and assuming the scheduling horizon of 1 week, discretized into 21 time intervals of 8 hours.

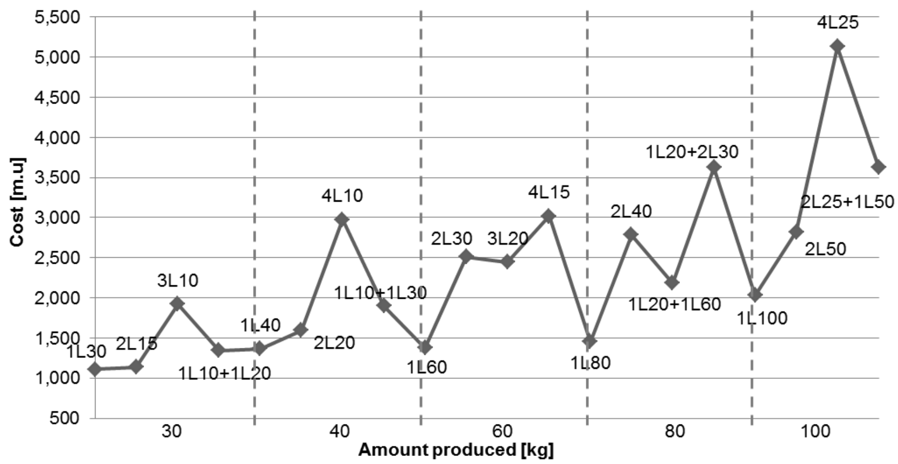


Figure 5.5 – Production lots cost in monetary units (m.u.) for product P1 (1L30 = 1 lot of 30 kg; 2L15 = 2 lots of 15 kg; and 1L10+1L20 = 1 lot of 10kg plus 1 lot of 20 kg).

Table 5.1 – Costs, number of reactors and capacity used and completion time for product P1.

Amount produced [kg]	Process/lots	Cost [m.u.]	Reactors capacity used [l]	Completion time [hours]
30	P1/1L30	1,108.6	15,600	112
	P1/2L15	1,134.6	15,600	128
	P1/3L10	1,922.5	27,100	152
	P1/1L10+1L20	1,341.5	19,300	88
40	P1/1L40	1,365.0	20,300	40
	P1/2L20	1,595.0	23,000	64
	P1/4L10	2,973.5	42,300	136
	P1/1L10+1L30	1,903.0	27,100	144
60	P1/1L60	1,375.0	20,300	40
	P1/2L30	2,511.8	35,900	144
	P1/3L20	2,446.0	34,500	88
	P1/4L15	3,021.6	42,300	136
80	P1/1L80	1,455.0	20,800	40
	P1/2L40	2,782.2	40,600	72
	P1/1L20+1L60	2,185.0	31,800	80
	P1/1L20+2L30	3,623.2	51,100	160
100	P1/1L100	2,030.0	28,600	88
	P1/2L50	2,823.7	40,600	72
	P1/4L25	5,131.2	71,400	168
	P1/2L25+1L50	3,631.2	51,100	152

[*] *1LOT30 = 1 lot of 30 kg; 2LOT15 = 2 lots of 15 kg; and1LOT10+1LOT20 = 1 lot of 10kg plus 1 lot of 20kg, etc;*

[**] *Reactors capacity used = total reactors capacity allocated to the process* $\sum_{k\in K_r}\sum_{l\in L_r}\sum_{t\in H}(V_{krl}^{max}N_{klt})$*, calculated during the post processing phase of CPLEX.*

When analyzing Figure 5.5 and Table 5.1, it can be seen that the production of 30 kg of product P1 has a cost always bellow 2,000. The most costly production case is to consider 3 lots of 10 kg (3L10), with a cost of 1,922.5, which leads also to the highest completion time. This indicates that the processing units are used inefficiently, as it can be seen by the value of 27,100 liters of reactors capacity allocated to the process (see Table 5.1). Assuming a production of 40 kg, the costs increase by 23% considering just one lot, however the completion time goes from 112 hours to just 40 hours. Again with the increase of the number of lots the process requires more reactors capacity and takes more time. An increase of the production to 60 kg (assuming a single lot) has almost no effect in the cost, and has no impact on the completion time and no impact on the reactors capacity allocated. Interestingly, the production of 3 lots of 20 kg results in a lower cost, in a lower reactors capacity and in a lower completion time, when compared with the production of 2 lots of 30 kg. The production of 80 kg (assuming a single lot) increases the costs by 6% of the 60 kg production case, but keeps the same completion time. While the production of 100 kg (assuming a single lot) increases the cost by 40% (assuming a single lot) and more than doubles the completion time, when compared with the 80 kg production case.

As conclusions it can be said that the tradeoffs associated to the scheduling decisions are related to the task-unit assignment and storage costs. The allocation of tasks to the smaller capacity units may result in lower operational costs, but may lead to longer completion times, since tasks may need to occur multiple times or a higher number of changeovers may be required; both situations potentially leading to an increase in the storage costs.

5.5.4.2 Scheduling Solutions

Following the previous analysis, we now derive full schedules for the production of the four products, with different lot sizes and scheduling horizons of 1 and 2 weeks. Since all products are scheduled simultaneously, the tradeoffs discussed above become more complex, resulting in the Gantt charts of Figure 5.6.

The first instance (INST1) is depicted in Figure 5.6 a) and considers the production of just one lot in a scheduling horizon of 1 week (21 time intervals of 8 hours). Figure 5.6 b) shows the second instance (INST2) also based on a single lot, but considering now a scheduling horizon of 2 weeks (42 time intervals of 8 hours). Figure 5.6 c) depicts the third instance (INST3) for the production of 2 lots of each product, in a scheduling horizon of 2 weeks. Table 5.2 summarizes the cost structure of each instance. The objective function utilized is the profit maximization $\max Z2$ (see expression (5.16)).

Table 5.2 – Cost structure of each instance.

Instance	Profit [m.u.]	Value of the products [m.u.]	Storage costs [m.u.]	Operational costs [m.u.]	Backlog costs [m.u.]	Lot costs [m.u.]
INST1	48,438.8	56,000.0	1,646.2	5,520.0	0.0	395.0
INST2[1)]	49,441.4	56,000.0	1,413.6	4,750.0	0.0	395.0
INST3[2)]	48,754.7	56,000.0	1,450.3	5,350.0	0.0	445.0

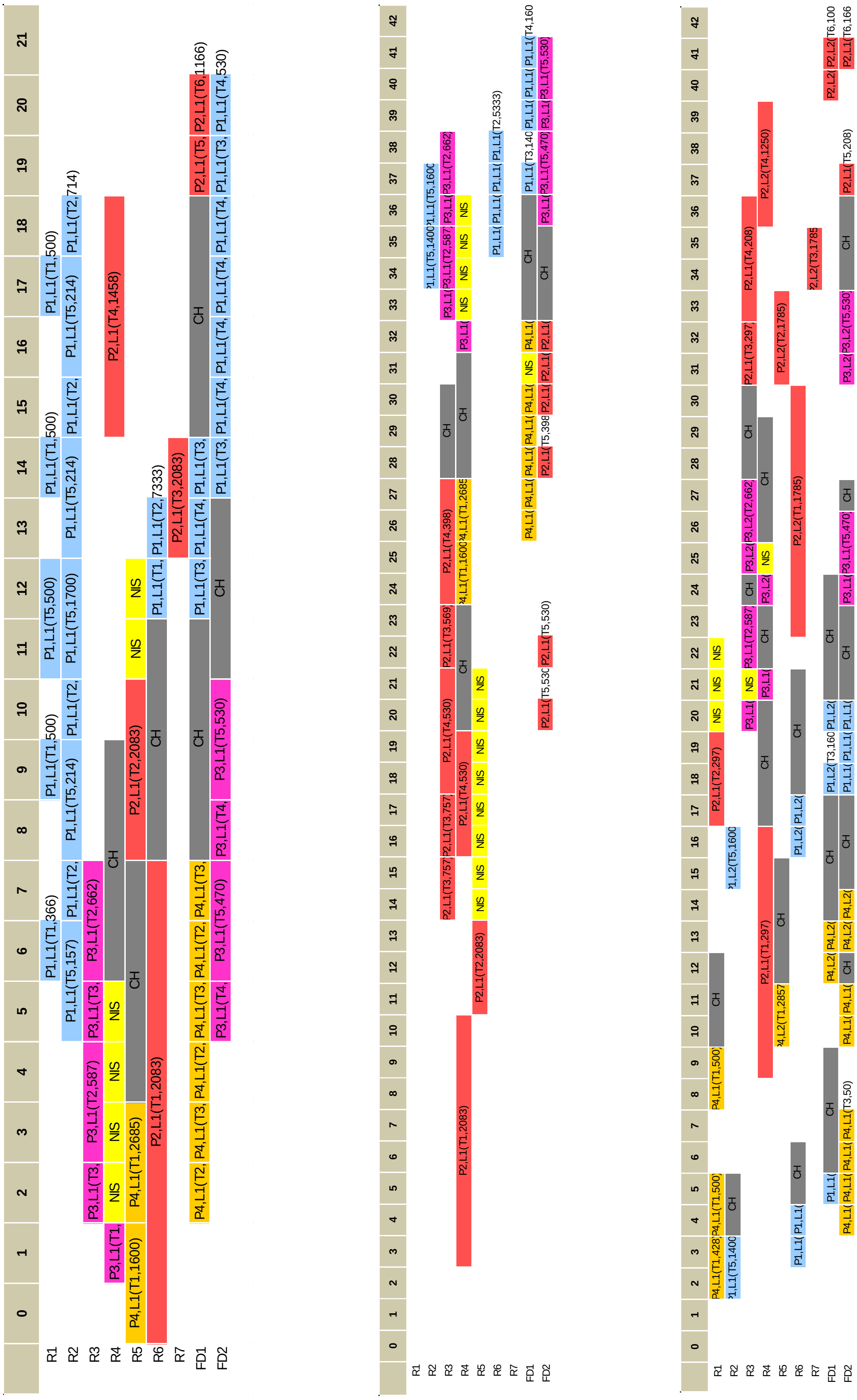
[1)] Solution within 3.32% of the optimal solution; [2)] solution within 5.34% of the optimal solution.

The most compact schedule is obtained with instance INST1 (see Figure 5.6). Processing units need to accommodate the demand in just one week, and this leads to a high occupation rate of the processing units. By extending the scheduling horizon to 2 weeks (INST2), units can be used more efficiently, with a reduction of the costs and an associated profit increase, see Table 5.2.

For INST1 the profit is equal to 48,438.8 m.u., while INST2 has a profit of 49,441.4 m.u., which is at least 2% higher, since the solution of INST2 has potentially some margin for improvement because it is not an optimal solution (3.32% gap). Moreover, the schedule of INST1 is inherently more complex to execute in practice, since several production tasks are repeated in order to fulfill the demand, see Figure 5.6 a). For example, in INST1 it can be seen that TASK1 of product P1 occurs 5 times, while in INST2, see Figure 5.6 b), this task occurs only 2 times.

In instance INST3, we have defined 2 lots for each product in a scheduling horizon of 2 weeks. This scenario tends naturally to impose additional idle periods for the units, as a consequence of the changeover periods; nevertheless the profit is comparable with the one obtained in INST1. Looking at the lot size decision variables of instance INST3, Product P1 has lots with 56 kg and 64 kg, resulting into a total amount of 120 kg. The demand of Product P2 was 70 kg, resulting into one lot of 16.8 kg and another of 53.2 kg. The lots of product P3 have 23.5 kg and 26.5 kg for fulfilling a demand of 50 kg, and product P4 had a demand of 60 kg that was fulfilled through lots of 28 kg and 32 kg. As discussed in the above section, the lot size has impact on the task-unit assignments, thus lots of the same product may have been assigned to different processing units. Globally, INST3 has lower storage costs, but has higher operational and lot setup costs.

Concerning the model performance, the computational time to obtain solutions is surely the main drawback when solving large instances. The numerical results demonstrate that CPLEX version 12.5.1 could not prove optimality for INST2 and INST3 during a computational time of 3,600 seconds. However, it should be noted that, from a practical perspective, the solutions presented in this paper have been considered very satisfactory by the industrial practitioners.



a) b) c)

Figure 5.6 – Production schedules: a) INST1; b) INST2; c) INST3.

5.6 Implementation

Although a huge progress has been recently achieved on the development of new mathematical formulations, conceptual frameworks and ontologies, the implementation of optimization models in the industry is far from being trivial. The implementation challenges are due to a wide spectrum of issues that are related to: a) understanding the model capabilities and limitations by the industrial practitioners; b) definition of model specifications and their impact on the decision-making process; c) definition of the most relevant modeling tradeoffs (model detail *versus* computational time *versus* quality of the solutions); d) development of efficient models capable to be used across several functions inside a company; e) models assessment; and f) models scale-up to the development of robust software applications.

The industrial context has motivated the present work and based on the results obtained, some guidelines are now provided that may help academics and industry practitioners on the collaborative development and deployment of optimization tools with industry. In fact our experience suggests that the implementation of optimization models in real production environments can strongly benefit from a previous development of case-specific models, oriented to a confined manufacturing system. These first developments provide academics and industrial practitioners with the necessary knowledge and confidence to address more complex problems. Thus in general, the implementation approach should go from case-specific to more general models.

Following a model-based approach, as suggested in this paper, the definition of the scheduling problem should be done together with the definition of the model specifications. This will contribute to the alignment of the problem requirements with the modeling capabilities.

Test and evaluation of models are also critical activities, since they provide valuable information for the identification of problem constraints and the development of solution approaches, targeting the improvement of the model performance. In this direction, generic and flexible ways of delivering model solutions are required. Powerful prototype visualization tools to enable a fast analysis of solutions can in general be built with a reasonable effort.

The solution methodology proposed in this paper takes into account some of these implementation challenges and aims at contributing to the development and deployment of optimization models in industry. A major advantage of the methodology comes from the fact that it is based on *components*, used as a way to build the adequate information context (*Problem representation* and *Decision-making process*) and thus designing consistent models. We believe that methodologies as the one proposed in this work can clearly lead to further improvements in the area.

5.7 Further Developments

Having presented a scheduling methodology that allows solving scheduling problems and having illustrated its application to a real case-study from the chemical-pharmaceutical, we now identify some major research topics that can contribute for the improvement of the developed methodology and the adoption of optimization models by the industry.

The major disadvantage of using exact models is undoubtedly the prohibitive computational time required to obtain schedule solutions of large instances. Therefore, the development and assessment of non-exact methods such as (meta) heuristics and decomposition approaches is a natural and promising research line. The main goal of such research is to decrease the computational time, while still obtaining satisfactory solutions.

Rescheduling features should also be addressed in an explicit way so as to account for unexpected changes in the demand and processing time delays.

One important aspect along the work developed is concerned with the problem representation. This turned out to be key point to guarantee close interaction with the planners. Although further developments on it should be performed namely, it should be improved in order to deal with more complex process operations and restrictions, and to enhance the cost modeling features. These functionalities can also be very useful to other participants in the scheduling process, such as sales and R&D departments.

Furthermore, any problem representation must be based on coherent information structures and this can be achieved by using standards such as ANSI/ISA (S88 and S95). The standardization of the solution methodology presented in this paper is a natural next step of this research, and will contribute to the harmonization of concepts and to the generalization of the information structures, thus facilitating the integration with other manufacturing systems. It is recognized (Henning, 2009; Klatt & Marquardt, 2009) that there is a lack of software packages capable of representing the scheduling problem in a user-friendly way, and that are not therefore immediately accessible to industrial practitioners. In this way, research should focus on the difficulties of the development and test of model-based approaches collaboratively with the industry. More efforts should be devoted to the development of confined optimization applications, in which the required data can be captured through a reduced number of steps and integrated in the decision-making processes of the companies. This will promote the development and assessment of optimization models in real production environments and consequently promote their adoption by the industry.

5.8 Conclusions

The existing planning and scheduling frameworks and ontologies provide a clear view on the typical requirements, information flows, core decisions and integration issues of the scheduling decision-making processes. They present building blocks for structuring complex and highly integrated systems. Nevertheless frameworks do not yet give an answer to the question of how planning or scheduling should be done in a specific company or industry. This happens because planning and scheduling decision-making processes are usually case-specific, and the available building blocks are not enough to define and integrate planning optimization methods in companies.

In this paper, we have proposed a solution methodology for production scheduling in chemical batch plants, supported by a MILP model. Our methodology has integrated some characteristics of existing frameworks, and was applied to a real case in the chemical-pharmaceutical industry, so as to build a systematic approach for representing and solving the scheduling problems.

We have developed a MILP discrete-time model based on the one proposed by Moniz et al. (2013), however other models could also be used. The data used by the MILP is automatically taken from the process representation tool developed in this work. The model is then run for different scenarios, and scheduling solutions and key performance indicators are represented in Gantt charts and tables, to support the decision-making process of the planners.

On the case-study addressed, two types of analysis were done. First, an evaluation of the processes alternatives and their associated costs was performed. Second, production schedules for scheduling horizons of 1 and 2 weeks were produced. Numerical results show that the model performed well in all instances. However CPLEX could not prove optimality for the larger problem instances. In general, the developed framework proved to be very useful for the company in the scheduling decision-making process and provided a solid base for structuring the scheduling related data.

The proposed methodology has a set of advantages, which are related to the general representation of the scheduling problem that can be used by several departments in the company and to the integration of the decision-making process with the optimization model. In our view, this has been a missing unifying point that could promote the adoption of planning and scheduling optimization tools in the industry. Methodologies should clearly define how tools should be applied and used in the company decision-making processes. In this field, research work is still required to map current planning and scheduling practices into coherent methodologies capable of efficiently using methods and tools, to systematically delivery planning and scheduling solutions.

The experience presented in this paper clearly shows the need for new innovative approaches and further levels of cooperation between academia and industry, to address the still open challenges in the adoption of advanced optimization approaches for industrial companies.

Appendix

Planning and Scheduling Frameworks

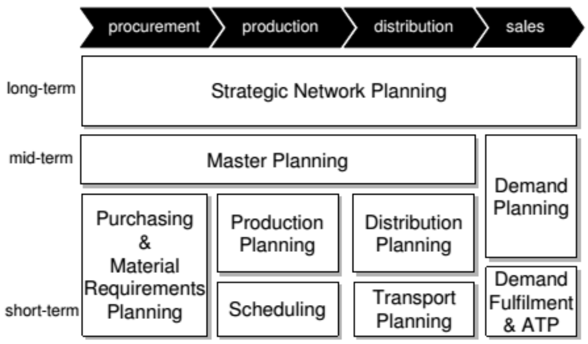


Figure 5.7 - Supply chain planning matrix - source (Meyr et al., 2005).

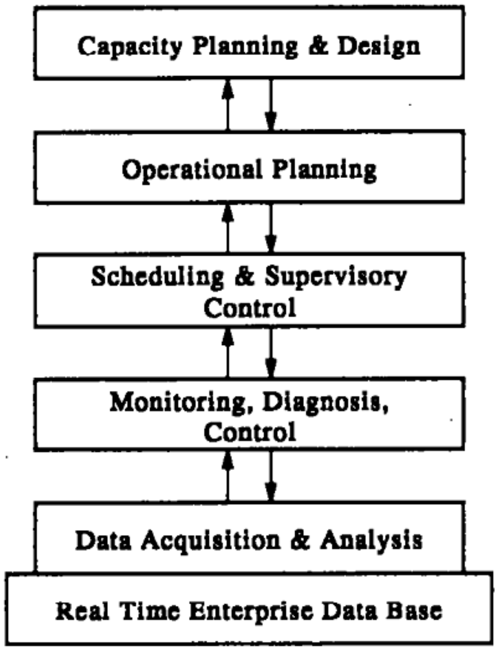


Figure 5.8 - Process operations hierarchy - source (Bassett et al., 1996)

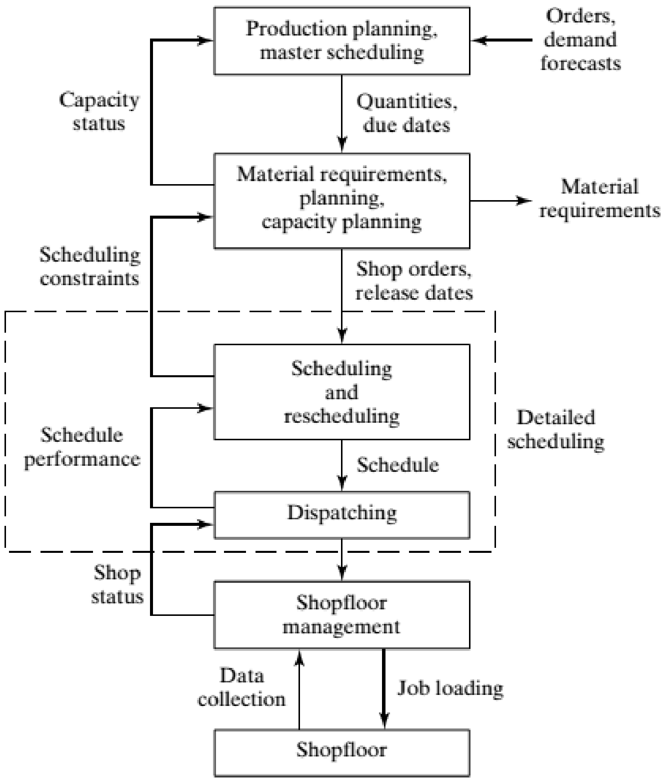


Figure 5.9 - Planning information flow in a manufacturing system – source (Pinedo, 2002).

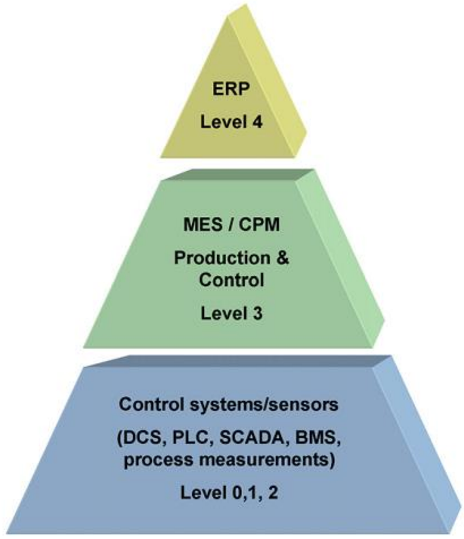


Figure 5.10 - Automation pyramid - source (Harjunkoski et al., 2009) .

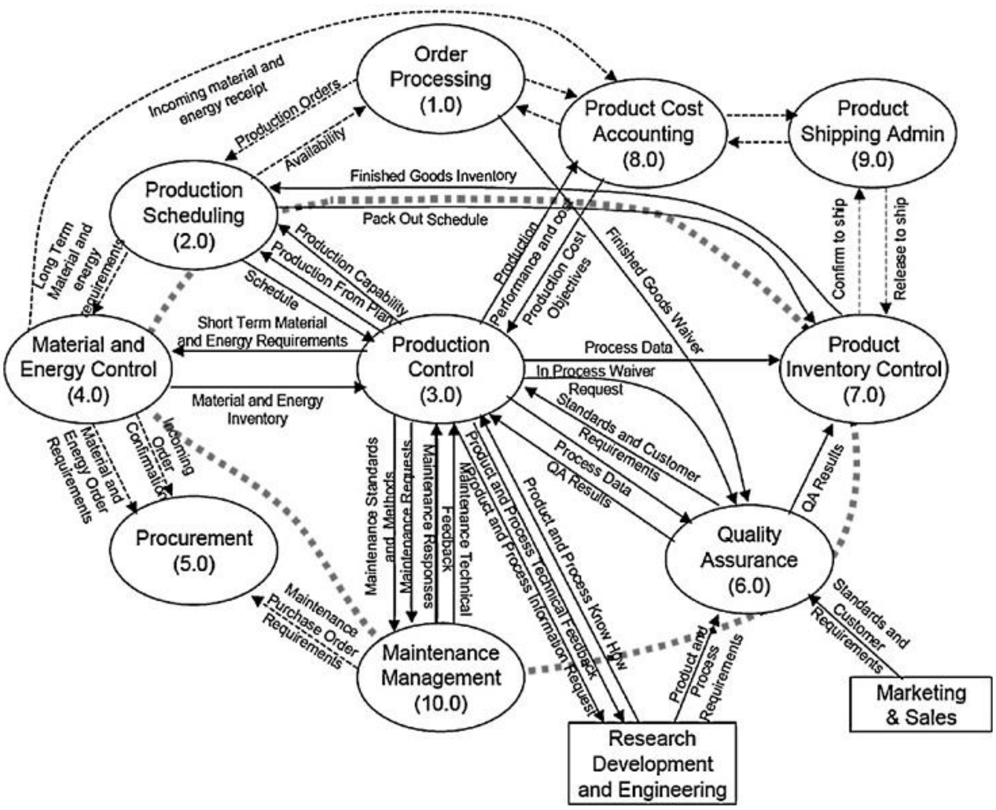


Figure 5.11 - Purdue Reference Model(ANSI/ISA-95, 2000).

Notation

Indices

d	delivery period
l	Lot
k	Task
p	Product
r	resource (processing unit, intermediary or final product)
t	time interval

Sets

D_{rt}	demand of product *r* at time interval *t*
DW_{rld}	delivery window of lot *l* and resource *r* (final product) at delivery period *d*
E	processing units
H	scheduling horizon
I	intermediaries
I^{NIS}	intermediaries subject to a non-intermediate storage policy
I_k^{NIS}	intermediaries produced by task *k* and subject to a non-intermediate storage policy
L	lots
L_r	lots associated with resource *r*
L_k	lots associated with task *k*
K_r	tasks that require resource *r* (processing unit, intermediary or final product)
K_r^c	tasks that consume resource *r* (intermediary or final product)
K_r^p	tasks that produce resource *r* (intermediary or final product)
P	products
R	production resources

Parameters

τ_k	processing time of task *k*

$\nu_{kr\theta}$, $\nu_{kr\theta}^{p}, \nu_{kr\theta}^{c}$	production/consumption proportion of resource (intermediary or final product) *r* in task *k* at time *Θ* relative to the start of task
c_{r}^{sto}	cost of storage of products and intermediaries *r*
c_{k}^{op}	operational costs of task *k*
c_{l}^{fix}, c_{l}^{var}	lot fix and variable costs
$c_{rll'}$	changeover time in processing unit *r* from lot *l* to lot *l'*
f_{l}^{r}	tasks *k* associated with processing unit *r* and lot *l*
q_{l}^{min}, q_{l}^{max}	minimum and maximum lot *l* size
R_{rt}^{max}	maximum resource availability of resource *r* (intermediary or final product) at time interval *t*
R_{krl}^{init}	resource *r* (intermediary or final product) availability of lot *l* at task *k* in the beginning of the planning horizon
T	length of the scheduling horizon
T_{ld}^{ed}	earliest time interval of lot *l* at delivery *d*
T_{ld}^{dd}	latest time interval of lot *l* at delivery *d*
v_{r}	value of product *r*
$V_{krl}^{min}, V_{krl}^{max}$	minimum and maximum capacity of resource *r* (processing unit) for task *k* of lot *l*

Variables

β_{l}	amount of product manufactured by lot *l* – lot size (continuous)
ξ_{klt}	batch size of task *k* and lot *l* at time interval *t* (continuous)
Π_{krlt}	delivery of resource (final products) *r* of lot *l* at time interval *t* available from task *k* (continuous)
N_{klt}	binary variables that are equal to 1 if task *k* starts lot *l* at time interval *t*
R_{krlt}	resource *r* (intermediaries or final products) availability, produced by task *k* of lot *l* at time interval *t* (continuous)
R_{krlt}^{c}	amount of resource *r* (intermediaries or final products) consumed from task *k* of lot *l* at time interval *t* (continuous)
R_{krlt}^{p}	amount of resource *r* (intermediaries or final products) produced by task *k* of lot *l* at time interval *t* (continuous)
Y_{l}	binary variables that are equal to 1 if *l* lot has been produced

Supporting Information

Networks

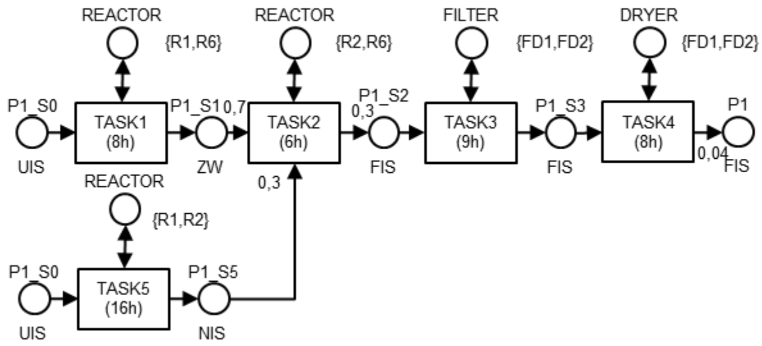


Figure 5.S1 – Recipe of product P1.

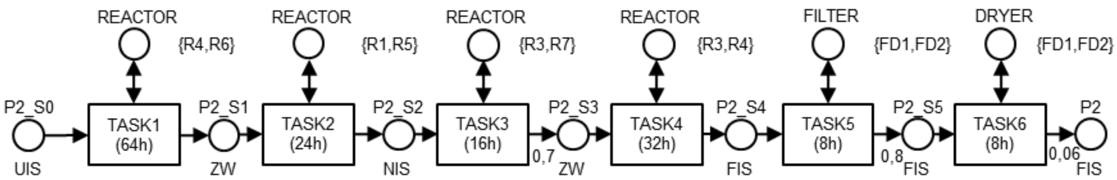


Figure 5.S2 - Recipe of product P2.

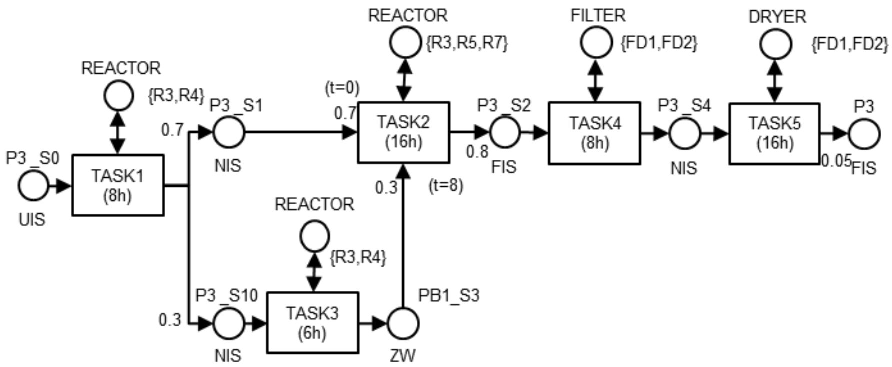


Figure 5.S3 - Recipe of product P3.

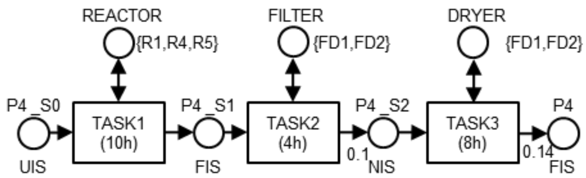


Figure 5.S4 - Recipe of product P4.

Data

Table 5.S1 – Products demand, delivery dates and backlog costs.

Product	Earliest delivery date	Latest delivery date	Minimum amount [kg]	Maximum amount [kg]	Backlog costs [m.u/kg]
P1	21	21	0	120	250
P2	21	21	0	70	200
P4	21	21	0	60	100
P3	21	21	0	50	120

Table 5.S2 – Minimum and maximum lot sizes, economic value and lot setup costs.

Product	Lot	Minimum lot size [kg]	Maximum lot size [kg]	Value [m.u]	Lot setup cost [m.u.]
P1	L1	10	80	250	5
P1	L2	10	80	250	5
P2	L1	10	60	200	25
P2	L2	10	60	200	25
P3	L1	10	30	120	10
P3	L2	10	30	120	10
P4	L1	10	40	100	10
P4	L2	10	40	100	10

Table 5.S3 – Materials initial, minimum and maximum availability, inventory costs and storage policy.

Resources	Init. availability [kg]	Max. availability [kg]	Inventory cost [m.u./kg]	Storage policy
P1_S0	100,000	100,000	0.07	UIS
P1_S1	0	0	0	ZW
P1_S2	0	10,500	0.04	FIS
P1_S3	0	4,500	0.01	FIS
P1_S5	0	0	0.08	NIS
P1	0	10,000	0.9	FIS
P2_S0	100,000	100,000	0.05	UIS
P2_S1	0	0	0	ZW
P2_S2	0	0	0.03	NIS
P2_S3	0	0	0	ZW
P2_S4	0	2,000	0.07	FIS
P2_S5	0	3,500	0.01	FIS
P2	0	10,000	0.6	FIS
P3_S0	100,000	100,000	0.05	UIS
P3_S1	0	0	0.03	NIS
P3_S2	0	3,000	0.04	FIS
P3_S3	0	0	0	ZW
P3_S4	0	0	0.07	NIS
P3_S10	0	0	0.02	NIS
P3	0	10,000	0.8	FIS
P4_S0	100,000	100,000	0.05	UIS
P4_S1	0	1,500	0.03	FIS
P4_S2	0	0	0.07	NIS
P4	0	10,000	0.5	FIS

Table 5.S4 – Characteristics of the processing units.

Unit	Type	Min. Volume [l]	Max. Volume [l]	Cost [m.u.]
R2	REACTOR	45	500	50
R3	REACTOR	75	1,700	100
R5	REACTOR	20	800	60
R6	REACTOR	70	2,700	200
R7	REACTOR	60	4,800	300
R8	REACTOR	150	9,300	600
R9	REACTOR	65	12,400	700
FD1	FILTER/DRYER	25	1,600	20
FD2	FILTER/DRYER	20	530	10

Table 5.S5 – Changeovers time of each unit (cr).

Unit	cr [hours]
R1	8
R2	16
R3	8
R4	16
R5	16
R6	16
R7	24
FD1	16
FD2	8

Table 5.S6 – Changeovers time between products (cr + products changeover time in hours).

Products	P1	P2	P3	P4
P1	c_r	c_r+2	c_r+2	c_r+16
P2	c_r+16	c_r	c_r+2	c_r+16
P3	c_r+16	c_r+16	c_r	c_r+16
P4	c_r+16	c_r+16	c_r+16	c_r

Table 5.S7 – Numerical results of the processes evaluation.

Amount produced [kg]	Process/lots	Int. variables/cont. variables/ Constraints	Nodes	Iterations	Gap (%)	Objective	CPU time (sec)
30	P1/1LOT30	221/1124/1858	70,183	578,331	0.0	1,108.6	5.8
	P1/2LOT15	442/2225/5005	20,871	379,326	0.0	1,134.6	10.7
	P1/3LOT10	663/3326/8372	29,256	1,479,559	0.0	1,922.5	41.4
	P1/1LOT10+1LOT20	442/2225/5005	35,102	538,112	0.0	1,341.5	10.6
40	P1/1LOT40	221/1124/1858	90,210	554,767	0.0	1,365.0	6.2
	P1/2LOT20	442/2225/5005	43,783	788,227	0.0	1,595.0	12.9
	P1/4LOT10	884/4427/11959	75,031	8,891,340	0.0	2,973.5	275.3
	P1/1LOT10+1LOT30	442/2225/5005	69,026	1,595,698	0.0	1,903.0	22.4
60	P1/1LOT60	221/1124/1858	210	2,901	0.0	1,375.0	1.1
	P1/2LOT30	442/2225/5005	72,327	2,298,281	0.0	2,511.8	30.0
	P1/3LOT20	663/3326/8372	35,729	1,668,960	0.0	2,446.0	56.8
	P1/4LOT15	884/4427/11959	49,107	3,358,538	0.0	3,021.6	139.1
80	P1/1LOT80	221/1124/1858	75	981	0.0	1,455.0	1.2
	P1/2LOT40	442/2225/5005	47,819	1,125,153	0.0	2,782.2	18.3
	P1/1LOT20+1LOT60	442/2225/5005	17,652	315,810	0.0	2,185.0	7.4
	P1/1LOT20+2LOT30	663/3326/8372	104,495	7,006,803	0.0	3,623.2	220.8
100	P1/1LOT100	221/1124/1858	27,578	295,301	0.0	2,030.0	4.0
	P1/2LOT50	442/2225/5005	1,544	588,788	0.0	2,823.7	12.8
	P1/4LOT25	884/4427/11959	40,646	5,071,307	0.0	5,131.2	236.5
	P1/2LOT25+1LOT50	663/3326/8372	46,179	4,404,857	0.0	3,631.2	144.9

Table 5.S8 – Numerical results of the production schedules.

Instance	Int. variables/cont. variables/ Constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
Inst1	884/4669/9350	1,256,627	96,031,323	52,768.9	0.01	48,438.8	1,474
Inst1[1)]	884/4669/9350	26,558	2,875,997	52,768.9	4.41	47,773.4	98
Inst2[2)]	1724/9121/18233	480,471	40,955,822	52,779.1	3.32	49,441.4	3,672
Inst2[1)]	1724/9121/18233	7,617	1,291,875	52,779.1	4.90	49,130.5	80
Inst3[2)]	3448/18069/52036	70,560	10,987,254	52,738.9	5,34	48,754.7	3,623

[1)] Stopping criterion is the integrality gap of 5%.; [2)] Stopping criterion is the time limit of 3,600 seconds.

References

Amaro, A., & Barbosa-Póvoa, A. (2008a). Planning and scheduling of industrial supply chains with reverse flows: A real pharmaceutical case study. Computers & Chemical Engineering, 32, 2606-2625.

Amaro, A., & Barbosa-Póvoa, A. (2008b). Supply chain management with optimal scheduling. Industrial & engineering chemistry research, 47, 116-132.

ANSI/ISA-88. (1995). Batch control part 1, models and terminology, (see also IEC 61512-01). Research Triangle Park, NC: ISA.

ANSI/ISA-95. (2000). Enterprise-Control System Integration. Part 1: Models and terminology.: ISA.

Applequist, G., Samikoglu, O., Pekny, J., & Reklaitis, G. (1997). Issues in the use, design and evolution of process scheduling and planning systems. ISA Transactions, 36, 81-121.

Barbosa-Povoa, A. P. (2007). A critical review on the design and retrofit of batch plants. Computers & Chemical Engineering, 31, 833-855.

Barbosa-Povoa, A. P., & Macchietto, S. (1994). Detailed design of multipurpose batch plants. Computers & Chemical Engineering, 18, 1013-1042.

Bassett, M., Dave, P., Doyle, F., Kudva, G., Pekny, J., Reklaitis, G., Subrahmanyam, S., Miller, D., & Zentner, M. (1996). Perspectives on model based integration of process operations. Computers & Chemical Engineering, 20, 821-844.

Castro, P., Barbosa-Povoa, A., & Matos, H. (2001). An improved RTN continuous-time formulation for the short-term scheduling of multipurpose batch plants. Industrial & engineering chemistry research, 40, 2059-2068.

Castro, P., Barbosa-Povoa, A. P., & Matos, H. A. (2003). Optimal periodic scheduling of batch plants using RTN-based discrete and continuous-time formulations: A case study approach. Industrial & engineering chemistry research, 42, 3346-3360.

Castro, P. M., & Grossmann, I. E. (2005). New continuous-time MILP model for the short-term scheduling of multistage batch plants. Industrial & engineering chemistry research, 44, 9175-9190.

Castro, P. M., Harjunkoski, I., & Grossmann, I. E. (2009). Optimal short-term scheduling of large-scale multistage batch plants. Industrial and Engineering Chemistry Research, 48, 11002-11016.

Engell, S., & Harjunkoski, I. (2012). Optimal operation: Scheduling, advanced control and their integration. Computers & Chemical Engineering.

Erdirik-Dogan, M., & Grossmann, I. E. (2008). Slot-based formulation for the short-term scheduling of multistage, multiproduct batch plants with sequence-dependent changeovers. Industrial & engineering chemistry research, 47, 1159-1183.

Erdirik-Dogan, M., Grossmann, I. E., & Wassick, J. (2008). Short-term scheduling of batch plants with parallel reactors forming mobile workgroups. Industrial & engineering chemistry research, 47, 6070-6080.

Floudas, C. A., & Lin, X. (2004). Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review. Computers & Chemical Engineering, 28, 2109-2129.

Grossmann, I. (2005). Enterprise-wide optimization: A new frontier in process systems engineering. AIChE Journal, 51, 1846-1857.

Grossmann, I. E., Caballero, J. A., & Yeomans, H. (1999). Mathematical programming approaches to the synthesis of chemical process systems. Korean Journal of Chemical Engineering, 16, 407-426.

Harjunkoski, I., & Grossmann, I. E. (2001). A decomposition approach for the scheduling of a steel plant production. Computers & Chemical Engineering, 25, 1647-1660.

Harjunkoski, I., Nyström, R., & Horch, A. (2009). Integration of scheduling and control—Theory or practice? Computers & Chemical Engineering, 33, 1909-1918.

Harjunkoski, I., Saliba, S., & Biondi, M. (2011). Production Optimization and Scheduling across a Steel Plant. In (Vol. 29, pp. 920-924).

Henning, G. P. (2009). Production scheduling in the process industries: current trends, emerging challenges and opportunities. Computer Aided Chemical Engineering, 27, 23-28.

Ierapetritou, M., & Floudas, C. (1998). Effective continuous-time formulation for short-term scheduling. 1. Multipurpose batch processes. Industrial & engineering chemistry research, 37, 4341-4359.

Janak, S. L., Lin, X., & Floudas, C. A. (2004). Enhanced continuous-time unit-specific event-based formulation for short-term scheduling of multipurpose batch processes: Resource constraints and mixed storage policies. Industrial & engineering chemistry research, 43, 2516-2533.

Kallrath, J. (2005). Solving planning and design problems in the process industry using mixed integer and global optimization. Annals of Operations Research, 140, 339-373.

Klatt, K.-U., & Marquardt, W. (2009). Perspectives for process systems engineering—Personal views from academia and industry. Computers & Chemical Engineering, 33, 536-550.

Kondili, E., Pantelides, C., & Sargent, R. (1993). A general algorithm for short-term scheduling of batch operations--I. MILP formulation. Computers & Chemical Engineering, 17, 211-227.

Kopanos, G. M., Méndez, C. A., & Puigjaner, L. (2010). MIP-based decomposition strategies for large-scale scheduling problems in multiproduct multistage batch plants: A benchmark scheduling problem of the pharmaceutical industry. European Journal of Operational Research, 207, 644-655.

Li, J., Xiao, X., Tang, Q., & Floudas, C. A. (2012). Production Scheduling of a Large-Scale Steelmaking Continuous Casting Process via Unit-Specific Event-Based Continuous-Time Models: Short-Term and Medium-Term Scheduling. Industrial & engineering chemistry research, 51, 7300-7319.

Li, Z., & Ierapetritou, M. (2008). Process scheduling under uncertainty: Review and challenges. Computers & Chemical Engineering, 32, 715-727.

Maravelias, C. T. (2012). General framework and modeling approach classification for chemical production scheduling. AIChE Journal.

Maravelias, C. T., & Grossmann, I. E. (2003). New general continuous-time state-task network formulation for short-term scheduling of multipurpose batch plants. Industrial & engineering chemistry research, 42, 3056-3074.

Maravelias, C. T., & Sung, C. (2009). Integration of production planning and scheduling: Overview, challenges and opportunities. Computers & Chemical Engineering, 33, 1919-1930.

Méndez, C., Henning, G., & Cerda, J. (2001). An MILP continuous-time approach to short-term scheduling of resource-constrained multistage flowshop batch facilities. Computers & Chemical Engineering, 25, 701-711.

Mendez, C. A., Cerda, J., Grossmann, I. E., Harjunkoski, I., & Fahl, M. (2006). State-of-the-art review of optimization methods for short-term scheduling of batch processes. Computers & Chemical Engineering, 30, 913-946.

Meyr, H., Wagner, M., & Rohde, J. (2005). Structure of advanced planning systems. In Supply chain management and advanced planning (pp. 109-115): Springer.

Moniz, S., Barbosa Póvoa, A. P., & Pinho de Sousa, J. (2013). New general discrete-time scheduling model for multipurpose batch plants. Industrial & engineering chemistry research.

Muñoz, E., Espuña, A., & Puigjaner, L. (2010). Towards an ontological infrastructure for chemical batch process management. Computers & Chemical Engineering, 34, 668-682.

Pacciarelli, D., & Pranzo, M. (2004). Production scheduling in a steelmaking-continuous casting plant. Computers & Chemical Engineering, 28, 2823-2835.

Pantelides, C. C. (1994). Unified frameworks for optimal process planning and scheduling. In (pp. 253-274): Cache Publications New York.

Pinedo, M. (2002). Scheduling : theory, algorithms, and systems. Upper Saddle, N.J.: Prentice Hall.

Pinto, T., Barbosa-Póvoa, A. P. F. D., & Novais, A. Q. (2005). Optimal design and retrofit of batch plants with a periodic mode of operation. Computers & Chemical Engineering, 29, 1293-1303.

Reklaitis, G. (1995). Scheduling approaches for the batch process industries. ISA Transactions, 34, 349-358.

Schilling, G., & Pantelides, C. (1996). A simple continuous-time process scheduling formulation and a novel solution algorithm. Computers & Chemical Engineering, 20, S1221-S1226.

Schulz, C., Engell, S., & Rudolf, R. (1998). Scheduling of a multi-product polymer batch plant: Citeseer.

Shah, N., Pantelides, C., & Sargent, R. (1993). A general algorithm for short-term scheduling of batch operations--II. Computational issues. Computers & Chemical Engineering, 17, 229-244.

Shaik, M. A., & Floudas, C. A. (2007). Improved unit-specific event-based continuous-time model for short-term scheduling of continuous processes: Rigorous treatment of storage requirements. Industrial & engineering chemistry research, 46, 1764-1779.

Stefansson, H., Sigmarsdottir, S., Jensson, P., & Shah, N. (2011). Discrete and continuous time representations and mathematical models for large production scheduling problems: A case study from the pharmaceutical industry. European Journal of Operational Research.

Stephanopoulos, G., & Reklaitis, G. V. (2011). Process systems engineering: From Solvay to modern bio-and nanotechnology.: A history of development, successes and prospects for the future. Chemical engineering science, 66, 4272-4306.

Sundaramoorthy, A., & Karimi, I. (2005). A simpler better slot-based continuous-time formulation for short-term scheduling in multipurpose batch plants. Chemical engineering science, 60, 2679-2702.

Sundaramoorthy, A., & Maravelias, C. T. (2011a). Computational Study of Network-Based Mixed-Integer Programming Approaches for Chemical Production Scheduling. Industrial & engineering chemistry research.

Sundaramoorthy, A., & Maravelias, C. T. (2011b). A general framework for process scheduling. AIChE Journal, 57, 695-710.

Susarla, N., & Karimi, I. (2010). Integrated Campaign Planning and Resource Allocation in Batch Plants. Computer Aided Chemical Engineering, 28, 1183-1188.

Till, J., Sand, G., Urselmann, M., & Engell, S. (2007). A hybrid evolutionary algorithm for solving two-stage stochastic integer programs in chemical batch scheduling. Computers & Chemical Engineering, 31, 630-647.

Venkatasubramanian, V., Zhao, C., Joglekar, G., Jain, A., Hailemariam, L., Suresh, P., Akkisetty, P., Morris, K., & Reklaitis, G. (2006). Ontological informatics infrastructure for pharmaceutical product development and manufacturing. Computers & Chemical Engineering, 30, 1482-1496.

Verderame, P. M., Elia, J. A., Li, J., & Floudas, C. A. (2010). Planning and Scheduling under Uncertainty: A Review Across Multiple Sectors. Industrial & engineering chemistry research, 49, 3993-4017.

Vooradi, R., & Shaik, M. A. (2012). Improved Three-Index Unit-Specific Event-Based Model for Short-Term Scheduling of Batch Plants. Computers & Chemical Engineering.

Wang, K., Löhl, T., Stobbe, M., & Engell, S. (2000). A genetic algorithm for online-scheduling of a multiproduct polymer batch plant. Computers & Chemical Engineering, 24, 393-400.

Wassick, J. M. (2009). Enterprise-wide optimization in an integrated chemical complex. Computers & Chemical Engineering, 33, 1950-1963.

Wassick, J. M., & Ferrio, J. (2011). Extending the Resource Task Network for Industrial Applications. Computers & Chemical Engineering.

6 Paper 4: Simultaneous Regular and Non-Regular Production Scheduling of Multipurpose Batch Plants

Samuel Moniz,[a,c] Ana Paula Barbosa-Póvoa,[a] Jorge Pinho de Sousa[b,c]

[a]*Centro de Estudos de Gestão, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal*

[b]*INESC TEC, 4200-465, Porto, Portugal*

[c]*Faculdade de Engenharia da Universidade do Porto, 4200-465, Porto, Portugal*

This chapter is under review in the journal Computers & Chemical Engineering.

Abstract

Regular and non-regular production can often be found in multipurpose batch plants, requiring two distinct operating strategies: campaign and short-term production. This paper proposes a solution approach for simultaneous scheduling of campaign and short-term products in multipurpose batch plants. Regular products follow a cyclic schedule and must cover several product deliveries during the scheduling horizon, while non-regular products have a non-cyclic schedule. The proposed approach explores the Resource-Task Network (RTN) discrete-time formulation. Moreover, a rolling horizon approach, and reformulation and branching strategies have been applied to deal with the computational complexity of the scheduling problem. Real case instances of a chemical-pharmaceutical industry are solved, showing the applicability of the solution approach.

Keywords: Multipurpose batch plants, campaign and short-term scheduling, rolling horizon, MILP models, Resource-Task Network.

6.1 Introduction

Multipurpose batch plants may operate in campaign or short-term modes, or may have both operational modes running in the same facility. The latter is the case of some process industries such as the chemical-pharmaceutical industry where high and low volume products need to be produced simultaneously. Products that are already in commercialization commonly present stable demands and are produced in large batches, being the campaign mode the preferred operational mode. In this case, the production resources are allocated to tasks that are executed in a cyclic way, thus defining production lines that tend to be stable for long periods of time. This approach leads to obvious benefits such as minimizing the changeovers costs while reducing the complexity of the operations. Alternatively, plants may also have short-term demands. Here, customers' orders of low quantities are placed for specific time windows. In the case of the chemical-pharmaceutical industry the products under development fit in this situation.

Globally, multipurpose batch plants need then to respond to a heterogeneous demand and plant resources have to be shared between campaign and short-term production modes. The plant responsiveness becomes critical and should be able to accommodate new orders at the minimum cost and with the minimum perturbation of the existing schedule, since as pointed out by Shah (2004), time-to-market is certainly the most important driver in the pharmaceutical industry.

Modeling and optimization methods have been extensively applied in batch processes problems, requiring a clever exploitation of the problem structure (Reklaitis, 1995). Moreover, the integration of different dynamic decisions such as design, planning and scheduling proved to be a good way of tackling the complexity of these problems (Barbosa-Povoa, 2007; Verderame et al., 2010).

The present paper addresses this problem and proposes a solution approach for scheduling multipurpose batch plants that simultaneously consider two different operating conditions – regular and non-regular production. The former encompasses products that are manufactured regularly in predefined production lines and the latter includes under development products having no defined production lines. The production resources are shared between both types of products. The rest of the paper is structured as follows. In section 6.2, a literature review is presented. The main characteristics of the scheduling problem are presented in section 6.3 and the proposed algorithm is described in section

6.4. In section 6.5, a mathematical model for the problem is presented. In section 6.6 are presented the solution methods and in section 6.7 the numerical results are discussed. Finally, in section 6.8 some concluding remarks are given.

6.2 Background

6.2.1 Scheduling of Multipurpose Batch Plants

Scheduling of multipurpose batch plants has been intensively addressed in the literature, covering a wide range of problems. On the production planning and scheduling problems a variety of modeling tools has been developed to tackle the associated problems, see reviews by (Mendez et al., 2006; Barbosa-Povoa, 2007; Li & Ierapetritou, 2008; Maravelias & Sung, 2009; Li et al., 2010; Verderame et al., 2010). Both the State-Task Network (STN) presented by Kondili et al. (1993) and the Resource-Task Network (RTN) suggested by Pantelides (1994) became two major frameworks used for solving scheduling problems in the chemical process industry, where discrete and continuous representations of time have been explored.

Discrete-time formulations easily model inventory and backlog costs, intermediate and delivery dates, and often result into compact formulations that can be easily modified. However, they present some problems when modeling variable processing times and sequence-dependent changeovers. Moreover, the efficiency of the discrete-time models and the feasibility of the solutions depend on the number and duration of the time intervals considered. To overcome these issues, continuous-time models were developed, where different time grids were used. Common time grid formulations for all resources were developed by (Schilling & Pantelides, 1996; Castro et al., 2001; Maravelias & Grossmann, 2003) and unit-specific time events formulations were developed by (Ierapetritou & Floudas, 1998; Janak et al., 2004; Vooradi & Shaik, 2012). Continuous time formulations lead however to more complex models and present larger integrality gaps. Even though the above mentioned developments represent a large step in the optimization of the process industry operation, the requirements found in real production environments often lead to new challenges that have not yet been adequately addressed in the literature.

One relevant scheduling issue regards the determination of detailed schedules in large time horizons. Such problem is due to various reasons. Scheduling problems may depend on recipes with short and long processing tasks, thus a sufficient large time horizon is required to accommodate all products. Moreover, production planning may need to be checked and validated at the operational level. These cases can be found in many chemical industries and are difficult to solve especially if different products recipes are present. The obvious and immediate approach for tackling this type of problems is to apply a short-term scheduling model for the entire planning horizon. However, a scheduling model with such dimension would hardly be solved. Alternative approaches such as cyclic scheduling, campaign planning and decomposition methods have been developed aiming at decreasing this modeling challenge.

Shah et al. (1993) presented a general framework for periodic scheduling of multipurpose batch plants. The model is based on the State-Task Network representation in which the “wraparound operator” is developed. This approach can deal with complex operations of batch plants, but it is only suitable for single campaigns. Later on, Schilling and Pantelides (1999) proposed a mixed integer non-linear programming (MINLP) model for addressing the periodic scheduling problem. Due to difficulties in the linearization the authors developed a special branch-and-bound (B&B) algorithm that branches the discrete and continuous variables. More recently, Pochet and Warichet (2008) propose a continuous time MILP formulation for solving the periodic scheduling problem and use strengthening techniques to improve the model computational time, and MIP based heuristic methods to obtain good solutions quickly in the larger instances. Addressing the same type of problem, You et al. (2009) compared the Dinkelbach’s algorithm with commercial MINLP solvers and verified that this algorithm performed better. Castro et al. (2003) proposed discrete and continuous-time formulations based on the RTN formulation for deriving optimal periodic schedules. Results favor the discrete-time periodic formulation in the case study addressed by the authors. Wu and Ierapetritou (2004) developed a cyclic schedule approach based on the STN using a continuous time formulation. This approach assumes stable demand for the time horizon under consideration and aims at determining the optimal cyclic schedule and cycle length. Moreover, the approach has a decomposition scheme for determining the startup and shutdown phases. Pinto et al. (2005) increased the complexity of the periodic scheduling by simultaneously considering the design and retrofit of multipurpose batch plants. The model is based on a discrete-time RTN formulation. Castro et al. (2008) solved an industrial scheduling problem from the chemical-pharmaceutical industry by proposing a periodic RTN formulation.

For campaign planning a number of algorithms have been presented in the literature. Mauderli and Rippin (1979) proposed a sequential approach where first alternative production lines with single products are generated, and then campaigns with several products are formed from the combination of two or more single product production lines. A screening procedure is applied to identify the dominant campaigns and, in a last step, a production plan is generated by solving a LP or MILP problem that allocates the dominant campaigns to the available production time. Papageorgiou and Pantelides (1993) proposed a hierarchical approach for multipurpose batch plants that takes into account the inherent flexibility of such plants with respect to intermediate storage policies and processing units utilization. A three-step approach is presented. The first step determines the number of campaigns and active stages in each campaign. The second step addresses the campaigns separately to derive the optimal cyclic schedules for the active stages and aims at improving the production rates of some stages. Finally, the third step reconsiders the timing of the campaigns determined in the previous step attempting to maximize the overall production value. Later Papageorgiou and Pantelides (1996a, 1996b) proposed a single-level model for planning and scheduling of multipurpose batch plants capable of simultaneously determining the campaigns (duration and products), the unit-task allocation and the task timings. Sundaramoorthy and Karimi (2004) propose a multi-period continuous-time MILP model. Computational tests have shown that the model is quite efficient even for long term planning periods. A limiting aspect of the approach followed by the authors is that production lines are considered instead of processing units, thus it is assumed that processing units are permanently allocated to a specific production line and cannot be shared. In practice, it is common to select a set of production resources to define a production line that will operate during a certain time period, sufficient to supply a given demand. More recently, Fumero et al. (2012) presented a solution approach for the scheduling of multistage multiproduct batch plants. They first solve a simplified slot-based continuous–time formulation that involves preordering constraints for the assignment of batches to slots in each stage. This provides a good upper bound for the campaign length of the detailed scheduling model solved in the second phase.

Regarding decomposition methods a good discussion is presented by Bassett et al. (1996). The authors analyze several decomposition methods for large-scale scheduling problems. Considering the same type of problems Wu and Ierapetritou (2003) developed an iterative approach that uses a lower bound obtained by heuristic-based decomposition approaches and an upper bound based on Lagrangean relaxation and Lagrangean decomposition. Lin et al. (2002) developed a rolling horizon approach. A two-level decomposition model is proposed to determine the current horizon and the products that shall be included. Wu and Ierapetritou (2007) also used a rolling horizon strategy to solve a planning and scheduling problem with uncertainty. A *sequence factor* is used to estimate the impact of the tasks sequencing in the planning problem. This parameter is used to make the planning and scheduling results converge. Erdirik-Dogan and Grossmann (2007) addressed the single stage problem with parallel units and sequence dependent changeovers. They propose an aggregate planning model that underestimates the effects of the changeovers and sequencing variables, but can be solved very efficiently; and a detailed scheduling model that models accurately the tasks sequencing and changeovers. To solve the larger instances a rolling horizon approach is suggested. Amaro and Barbosa-Póvoa (2008, 2008b) also studied the large scale scheduling and planning problems of batch plants using an extended STN representation (Chain-STN) in a supply chain context. An hierarchical decomposition procedure was proposed to link the planning with the scheduling decisions and a real case-study of a pharmaceutical industry was solved. Stefansson et al. (2011) proposed a decomposition algorithm that prioritizes the scheduling of the bottleneck units. The approach is applied to a multistage batch plant and the problem is decomposed into two parts. They start by solving the bottleneck stage and then solve the remaining stages. Moreover, they compare discrete-time formulation based on Kondili et al. (1993) with a continuous-time general precedence formulation based on Méndez et al. (2001). The continuous-time formulation (limited to sequential processes) have provided more accurate solutions and used less computational time, compared with the discrete-time general formulation. Recently, Sundaramoorthy and Maravelias (2011) shown that discrete-time models have many advantages over continuous-time formulations. Their study indicates that discrete-time models have better performance concerning the solution times and integrality gap. Moreover, discrete-time formulations can be easily modified to account for other processing characteristics. In order to address the computational burden of the MILP models, Velez and Maravelias (2013) propose three reformulations to define the number of batches of each task and use as basis the STN formulation. Tests have shown that branching on the integer variable number of batches eliminates many symmetric solutions, leading to improve the model performance.

The existing approaches can deal with several problem complexities, but they are still quite limited in simultaneously managing mixed operating strategies such as the regular and non-regular production, or the campaign and short-term scheduling. The work described in this paper aims at reducing this gap and proposes a simple three-step approach that tries to explore the specific problem structure and the current industrial planning procedures, as used in the chemical-pharmaceutical industry.

6.2.2 Motivation for a Mixed Strategy

Depending on the product recipes structure and on the allowable task / processing unit assignment, we may have multiproduct or multipurpose operating strategies (Reklaitis, 1995). Multiproduct batch plants are settled to manufacture products that have similar recipes, with production lines employing many-to-one processing unit / task assignments and operating cyclically to accommodate serial campaigns. Multipurpose batch plants under campaign operation are more appropriate for products with dissimilar recipes, allowing many-to-many processing unit / task assignments, and possibly having several campaigns involving several production lines, each operating cyclically. General multipurpose plants can also be defined and refer to multipurpose plants that operate with no defined production lines and with non-periodic production, where different types of products are simultaneously produced.

In practice, a mixed strategy may be present in a given plant. This occurs when the product portfolio combines characteristics of both strategies. In these cases, part of the plant may operate using dedicated production lines, while the other part operates in the multipurpose mode; or the same resources may be shared among the processes that have to be performed. Since production resources such as processing units, raw materials or utilities are shared, scheduling integration is required. Table 6.1 summarizes the characteristics of regular and non-regular production for the case of the chemical-pharmaceutical industry. Non-regular products are products that do not have long-term demand, they are produced in relatively low quantities and for specific time windows. Therefore, they do not justify the establishment of dedicated production lines. The pharmaceutical products under development fit into this category. On the other hand, regular products have typically well-defined recipes and stable production lines and involve the delivery of large amounts of products during long periods of time.

Table 6.1 – Characteristics of regular and non-regular production in the chemical-pharmaceutical industry.

Non-regular production	Regular production
There are stable and unstable product recipes. Recipes may change as a result of the process development.	Products have stable product recipes. Changes in the recipes are possible to do, but require legal and customer approvals.
Production has an irregular demand pattern and is triggered by customer orders.	Production has a regular demand pattern, usually established by a master production plan.
Demand needs are specified for a short period of time, typically few weeks.	Demand needs are planned for long term, typically from several months to one year.
Assignment of processing units to tasks can vary (*e.g.* scale–up of the production processes)	Assignment of processing units to tasks tends to be permanent, despite the existence of alternative processing units.
Products have tight delivery windows.	Products have relaxed delivery dates.

6.3 Problem Description

In the problem under investigation we consider that the following information is available: (i) the detailed recipes of the products that will be produced in campaign and short-term modes; (ii) the maximum and minimum capacity of the processing units; (iii) the demand and the delivery dates; (iv) changeover times required to clean units between products; (v) the minimum and the maximum cycle time for the regular products; and (vi) the costs and economic value of the products.

The objective is to maximize the overall profit for all products, while determining: (i) the cycle time *T* for the products to be scheduled in a campaign mode; (ii) the task unit assignment and sequencing for all products; (iii) the tasks batch sizes and storage levels; and (iv) the number of campaign cycles. Each product is defined by a recipe (see Figure 6.1) that identifies the task sequence with the respective processing time and allowable processing units. Raw materials and final products have unlimited storage, while storable intermediaries have finite storage. Task batch sizes are limited by the capacity of the processing unit chosen.

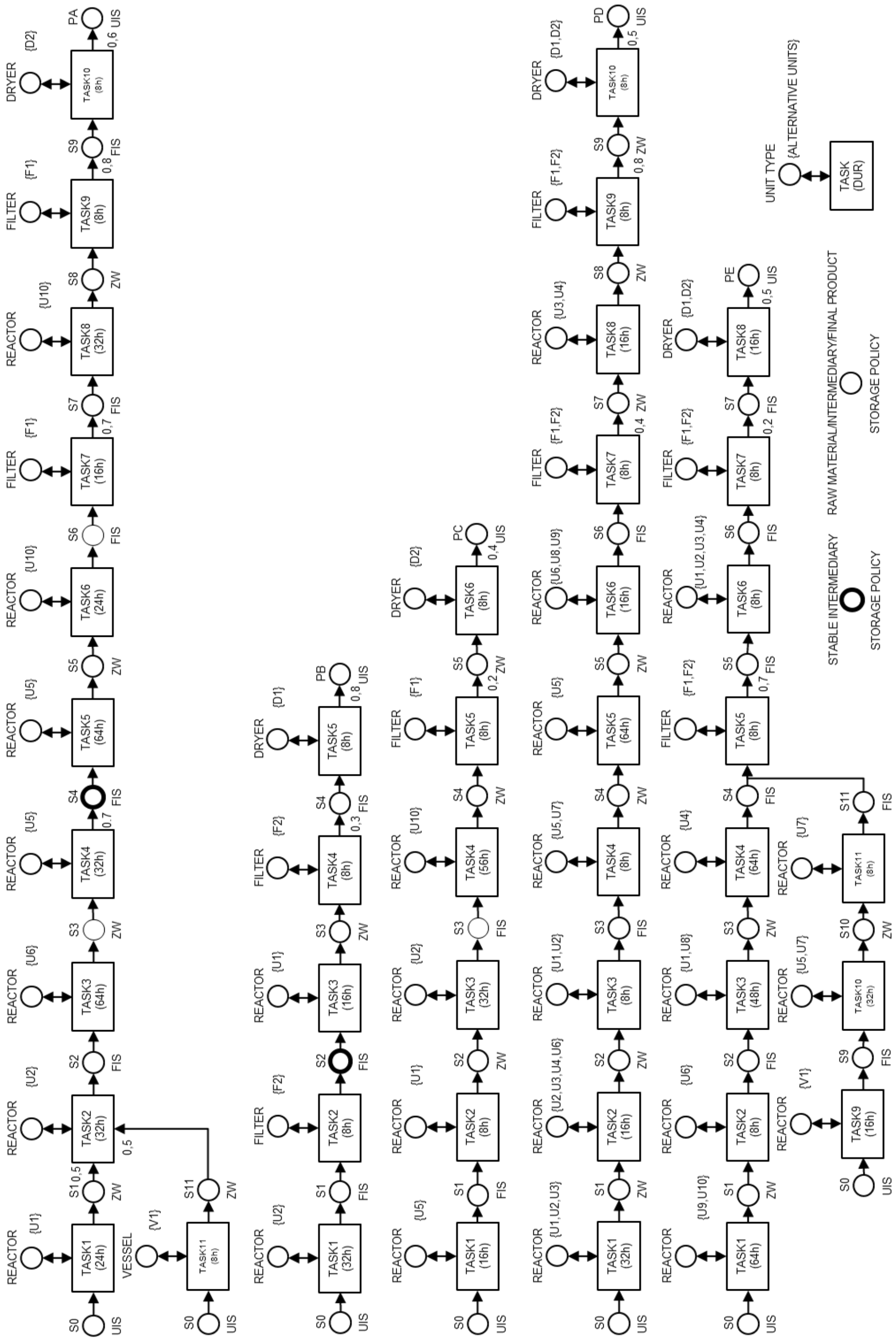


Figure 6.1 – Recipes of products PA, PB, PC, PD and PE.

6.4 Proposed Algorithm

To solve the integrated problem as described above, we propose the following three-step procedure (see Figure 6.2). In the first step, we determine the campaign schedule for the products that will be manufactured in campaign mode, with regular products being distributed into campaigns that may have one or more products. In the second step, we create campaign tasks for each schedule determined in the first step. These are aggregate tasks that consume and produce resources according to the campaign schedule. In the third step, we run the scheduling model having the campaign tasks of the regular products and the detailed recipes of the non-regular products. Campaign tasks follow the concept of supertasks that were firstly suggested by Zentner et al. (1994) and Bassett et al. (1996), and more recently by Moniz et al. (2012).

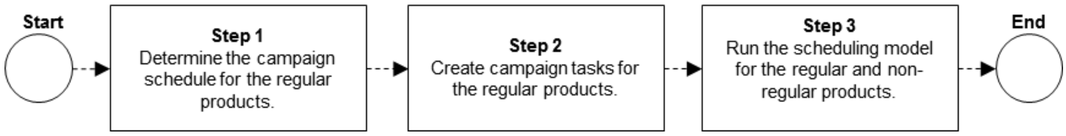


Figure 6.2 – Suggested approach for regular and non-regular production scheduling.

6.4.1 Step 1 – Determination of the Campaign Schedule

One possible approach to derive the campaign schedule is to run a periodic schedule formulation as suggested by Shah et al. (1993). In this case, the periodic schedule consists in a plan in which tasks are executed with a cyclic pattern. Processing units will have a cyclic operation as well. This schedule can be repeated successively until the demand is satisfied, assuming that periodic schedules can be derived and applied during a long time horizon, under stable operation and product demand (Shah et al., 1993). Cyclic scheduling implies the existence of two distinct time periods: the startup and shutdown phases. The former is related to the initial schedule that produces the intermediaries needed for the periodic schedule and the latter is related to the final schedule required for the conclusion of the production of all remaining intermediaries.

In practice, the periodic scheduling is a valid approach under the following assumptions. In the cases where the products are produced during long time horizons and the schedule can be replaced by a shorter and cyclic schedule that is repeated until the fulfillment of the demand. Products should have well defined recipes and few alternative production routes. If recipes have several alternative production routes, the tasks-units assignment will tend to vary as well. Therefore, it would be preferable to derive schedules in which the task-unit assignment is not limited to a repeating pattern initially determined. In the practical case of recipes having several alternative units per task, it would be desirable to use the task-unit assignment flexibility across the schedule, instead of using a repeating pattern during the entire time horizon.

Several advantages can be pointed out to the periodic scheduling. The suggested schedule is easier to implement due to the repetitive pattern of the tasks execution. Moreover, the computational burden of solving a large and non-periodic scheduling problem can be avoided by solving a smaller periodic scheduling problem.

To overcome the assumption of stable production demands and to avoid startup and shutdown effects an alternative approach is used in this work. A non-cyclic schedule can be derived assuming that the storable intermediaries are available in the beginning of the schedule execution and that are replaced in the same quantity when the schedule finishes. This schedule can be modeled using a campaign task that can be repeated successively during the scheduling horizon to satisfy the product demand. This allows the execution of campaign tasks without the need of startup and shutdown periods, being the overall schedule more responsive. The major disadvantage of this approach is the fact that storage costs of the intermediaries tend to be higher, this representing a tradeoff between schedule responsiveness and storage costs. Note that, although the schedule formulation being used is non-cyclic, this approach retrieves schedules in which tasks are executed with a specific cycle, thus we can still call this a cyclic schedule. This concept is better explained later on.

The storable intermediaries are specific characteristic of each recipe since they depend on the chemical stability and storage conditions of the material. For example, in Figure 6.1, the storable intermediaries are identified by the bold states.

As it can be seen in Table 6.1, the characteristics of regular production are appropriate for a cyclic schedule operation. Regular products have stable recipes, the assignment of the processing units to tasks tends to be permanent, the demand is known in advance for a long time horizon, and delivery dates are more flexible if compared with non-regular production, suggesting that this kind of production can be managed through a make-to-stock policy.

In this context, the most common objective function is the maximization of the net production over the cycle time under consideration. However, objectives such as maximizing the average profit or minimizing costs can be used too. In this work, we have used net production as the objective because we have assumed that campaigns have single products and that the control over the schedule production rate is a relevant indicator for measuring the schedules performance.

6.4.2 Step 2 – Creation of the Campaign Tasks

The level of abstraction chosen for modeling recipes will have a direct impact on the model size and therefore on its applicability. High detailed recipes will conduct to more exact models that are however more difficult to tackle computationally. On the other hand, less detailed recipes result into simpler models, and those are easier to handle. The strategy followed in this work exploits the problem structure as described above. Thus, regular products are modeled using campaign tasks. These are aggregate tasks that model the cycle-time and resources allocation/release profile of the schedule determined in step 1. Instead of having a detailed schedule that considers all resources and tasks, we have created a single task for modeling the entire schedule. In this way, many resources and tasks that are considered in step 1 can be disregarded. This reduces the model size in step 3, in terms of the number of binary and continuous variables and constraints.

For example, product PB requires five tasks and has a total of four intermediaries (see Figure 6.3), and from these only the intermediary S2 is storable. Using a cyclic schedule formulation any schedule having a cycle time equal to four (T=4) will serve the purpose (see Figure 6.3 a)). Nevertheless, the implementation of this schedule requires startup and shutdown phases, as shown in Figure 6.3 b). On the contrary, using the non-periodic formulation presented in section 6.5.1, startup and shutdown phases are not required, since materials availability is ensured by the campaign task (see Figure 6.3 c)). In this way, to produce two batches of product PB the cyclic scheduling requires 13 time intervals, while if campaign tasks are used 9 time intervals are sufficient.

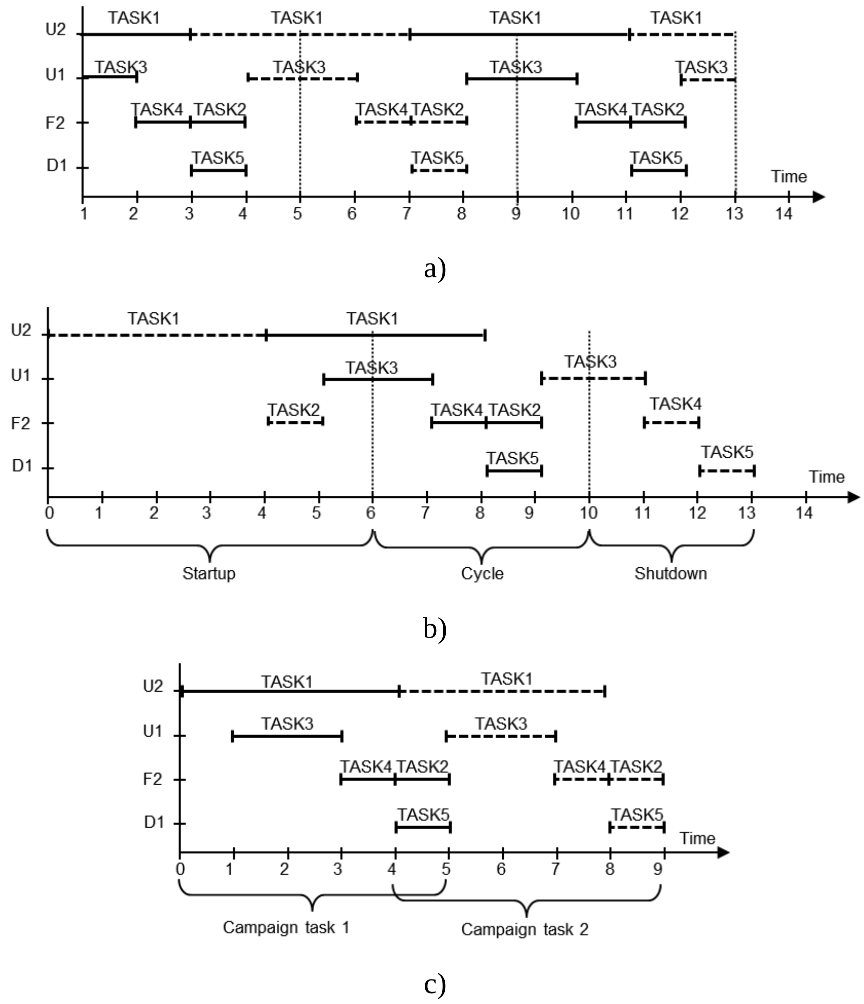


Figure 6.3 – a) cyclic schedule of PB (T=4); b) cyclic schedule of PB considering the startup and shutdown phases; c) scheduling of PB considering campaign tasks.

The corresponding campaign task is depicted in Figure 6.4 and will have a length of 5 time intervals ($\theta = 5$). Unit U2 is allocated to task TASK1 at the beginning of the campaign task execution ($\theta = 0$). Task TASK3 is executed one time interval after in unit U1 and consumes the previously stored intermediary S2, and at ($\theta = 4$) tasks TASK2 and TASK5 are executed to replenish the intermediary S2 and to produce product PB at ($\theta = 5$), respectively. It can be verified that the cycle of the periodic scheduling is 4 shifts and that the campaign task takes 5 shifts. However, since the campaign task allows for superposition of 1 shift the resulting throughput time of product PB is also equal to 4 shifts. The superposition of campaign tasks is then allowed as can be seen in Figure 6.3 c) and is defined in the mathematical formulation through the coefficient units' allocation/release of the RTN formulation.

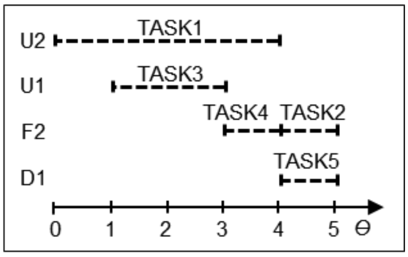


Figure 6.4 - Campaign task for product PB, with respective resource allocation profile.

This approach does not imply any reduction of the solution space of the schedule obtained in step 1, since processing units allocation/release to/from tasks is transposed to the campaign task respecting the sequencing obtained. Moreover, it is important to make a distinction between materials that need and do not need inventory control. If we need to have control over the availability of certain materials, for example stable intermediaries or final products, then these materials need to be modeled in the campaign task. These materials must be storable, to allow the execution of the campaign task without the need of the startup and shutdown phases. All the other materials can be omitted from the campaign task because they are produced and/or consumed within the campaign tasks, and we can simply assume that they are available when required.

6.4.3 Step 3 – Scheduling Model

Finally, the scheduling model in step 3 integrates both production types but with different aggregation levels. Regular production is modeled by campaign tasks, while non-regular production is represented by the detailed recipes. The model used in this step is presented in section 6.5.3.

The approach suggested in this work addresses the complex modeling challenge of the scheduling problem, by proposing different decomposition schemes, for different production types, that are typically found in the chemical-pharmaceutical industry.

6.5 Mathematical Formulations

The integrated algorithmic approach described above is based on a set of mathematical formulations that are characterized below.

6.5.1 Step 1 – Determination of the Campaign Schedules (CS model)

The campaign schedules are obtained using the non-periodic RTN discrete-time formulation (6.1) to (6.8).

By assuming that the storable intermediaries are replenished until the end of the schedule (time T) through constraints (6.4), the resulting schedule can be repeated successively and startup and shut-down phases can be avoided. The production resources R_l include processing units, raw materials, intermediaries and final products ($R_l = E \cup W_l \cup I_l \cup P_l$), where l is the campaign task ($l = 1, \dots, L$). In this way, we can define campaign tasks having different products that share the set of processing units E. The availability of the production resources is given by the resources balance constraints (6.1). R_{rt} are continuous variables that denote the availability of the resource r at time interval t, while N_{kt} are binary variables that are equal to one if task k starts at time interval t. The amount of resource (processing units) allocated or released by each task is specified by the parameter $\mu_{kr\theta}$, which can take values during the processing time of the task (τ_k). Similarly, materials are consumed and produced at the proportion $\nu_{kr\theta}$ of the task batch size that is modeled through the continuous variables ξ_{kt}. The resources maximum availability is guaranteed by constraints (6.2). Task batch size variables ξ_{kt} are activated through the binary variables N_{kt} in constraints (6.3), which also ensure that the task batch sizes are within the capacity limits of the processing units. The set of constraints (6.4) ensures the intermediaries balance at the end of the schedule. These constraints are essential to guarantee the replacement of the storable intermediaries at the end of the schedule. The net production of the final products over time T is given by constraints (6.5), where Δ_r is the net production of final product r. And constraints (6.6) define the production bounds for Δ_r, by imposing minimum and maximum amounts Θ_r^{min} and Θ_r^{max}, respectively. The variables domain is defined in (6.7).

Constraints

$$R_{rt} = \left(R_r^{init}|_{(t=0)}, R_{r,t-1}|_{(t>0)}\right) + \sum_{k \in K_r} \sum_{\theta=0}^{\tau_k} \left(\mu_{kr\theta} N_{k,t-\theta} + \nu_{kr\theta} \xi_{k,t-\theta}\right) \quad \forall r \in R_l, t \in H \tag{6.1}$$

$$0 \leq R_{rt} \leq R_{rt}^{max} \quad \forall r \in R_l, t \in H \tag{6.2}$$

$$V_{ke}^{min} N_{kt} \leq \xi_{kt} \leq V_{ke}^{max} N_{kt} \quad \forall e \in E, k \in K_e^l, t \in H \tag{6.3}$$

$$R_r^{init} = R_{r,T} \quad \forall r \in I_l \tag{6.4}$$

$$\Delta_r = R_{r,T} - R_{r0} \quad \forall r \in P_l \tag{6.5}$$

$$\Theta_{\mathrm{r}}^{min} \leq \Delta_r \leq \Theta_{\mathrm{r}}^{max} \quad \forall r \in P_l \tag{6.6}$$

$$\begin{aligned} &R_{rt} \in \mathbb{R}_+ \quad \forall\, \mathrm{r} \in R_{\mathrm{l}}, t \in H \\ &\xi_{kt} \in \mathbb{R}_+ \quad \forall k \in K_l, t \in H \\ &N_{kt} \in \{0,1\} \quad \forall\, k \in K_l, t \in H \\ &\Delta_r \in \mathbb{R}_+ \quad \forall r \in P_l \end{aligned} \tag{6.7}$$

Selection of the intermediaries

The formulation presented above assumes that the storable intermediaries are given so as to avoid startup and shutdown phases of the campaigns. Alternatively, the storable intermediaries can be determined by the optimization model, assuming that these intermediaries are not raw materials and final products, and that they have an initial amount that is replenished at the end of each campaign. The CS model can be easily modified to account for these requirements. The initial amount of the intermediaries is now given by the decision variable $R_r^{init'}$ that replaces the parameter R_r^{init} at constraints (6.1) and (6.4). For processing units, raw materials and final products $R_r^{init'}$ must be equal to the initial availability R_r^{init}, as expressed by (6.7.1) . For the intermediaries, $R_r^{init'}$ is confined by the maximum availability R_r^{max}, see (6.7.2). Note that, if R_r^{max} is equal to 0 then this intermediary is not eligible to be a storable intermediary.

$$R_r^{init'} = R_r^{init} \quad \forall r \in E \cup W_l \cup P_l \tag{6.7.1}$$

$$0 \leq R_r^{init'} \leq R_r^{max}.\lambda_r \quad \forall r \in I_l \tag{6.7.2}$$

$$\sum_{r \in I_l} \lambda_r \leq a \tag{6.7.3}$$

To have a limit on the number of storable intermediaries a new binary variable λ_r can be used. So, λ_r is equal to 1 if intermediary r has been selected as storable intermediary. Constraints (6.7.3) ensure that no more than a intermediaries can be storable.

Objective Function

The objective function is the maximization of the production rate and is given by expression (6.8). Several alternative schedules can be derived by solving the same model for different values of *T*, with *T* between *Tmin* and *Tmax*. *Tmin* is equal to the maximum processing time required to produce the stable intermediaries or the final product. *Tmax* is defined as the maximum acceptable duration for the campaign schedule. The selected schedule will give the maximum production Δ_r for each product r. Moreover, in order to calculate the minimum amount of product that can be delivered by each campaign cycle, the model was solved fixing the binary variables N_{kt} determined previously and assuming a minimization version of the objection function (6.8). The minimum and maximum values of Δ_r represent the production bounds of product r, and are used in step 2 as the minimum and maximum lot size (L_{rk}^{min}, L_{rk}^{max}) of product r at campaign task k.

$$\max \frac{1}{T} \sum_{r \in P_l} \Delta_r \tag{6.8}$$

6.5.2 Step 2 – Creation of the Campaign Tasks

Campaign tasks now are created taking as a basis the time T chosen in Step 1. These tasks will consume/allocate and produce/release resources according to the resources/tasks assignment made in Step 1. This approach allows modeling campaigns, as they are viewed as single production tasks, taking advantage of the uniform representation of the RTN formulation. Figure 6.5 depicts the campaign tasks for regular products PA and PB. The lot size is between the maximum (L_{rk}^{max}) and minimum (L_{rk}^{min}) allowable production taking into account the requirements of the recipes. The consumption and production proportions of the materials in the campaign tasks are calculated through the ratio *amount of material required* /*amount of final product.* Therefore, the campaign task of PA has a processing time of 152 hours which results in a net production of 235 kg at the maximum lot size and PB campaign task has a processing time of 40 hours and delivers 120 kg. In step 3, it is used a RTN non-periodic formulation for scheduling all products. In order to account for lot-size-dependent processing times and also alternative units, a piecewise approximation can be done by creating multiple instances of the campaign tasks. The new campaign tasks will have different lot-size intervals that correspond to different processing times and/or units.

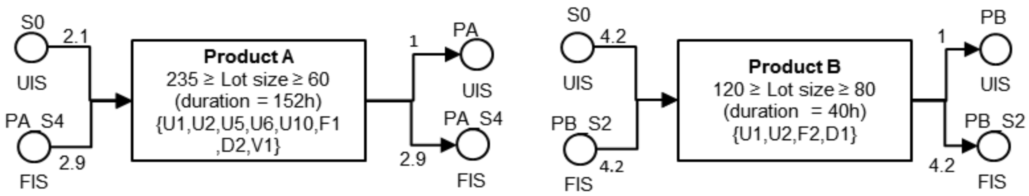


Figure 6.5 – Campaign tasks for the regular products PA and PB.

6.5.3 Step 3 – Detailed Scheduling Model (DS model)

Finally, a single schedule with campaign and short-term products is built by using constraints (6.9) to (6.18) and objective function (6.19). Again, we use as basis the RTN formulation.

In order to model sequence-dependent changeovers, the product index p is considered in the resource availability R_{pet}^{DS} continuous variable. Thus, R_{pet}^{DS} variables give the processing unit e availability for product p at time interval t. The changeover tasks are defined by $C_{epp't}^{DS}$ binary variables that are equal to 1 if a changeover task occur on the processing unit e between products p and p' at time interval t. The assignment/sequencing N_{kt}^{DS} variables and the batch size ξ_{kt}^{DS}variables are similar to the CS model. The superscript DS in the variables and sets indicate that they refer to the detailed model.

In this way, the resources balance constraints (6.9) determine the availability of the processing units for each product and time interval. The unit availability R_{pet}^{DS} is equal to the availability in the previous time interval $R_{pe,t-1}^{DS}$ plus the availability resulting from the unit's allocation/release to/from the production or changeover tasks at time interval t. The production tasks coefficients $\mu_{ke\theta}$ define the unit e allocation/release done by task k at time θ relative to the start of the task, and the changeovers coefficients $\alpha_{ep'p''p\theta}$ give the allocation/release of unit e from product p' to p'' being the product p held by the unit e at time θ relative to the start of the changeover task. Constraints (6.10) do the initial assignment of processing units to products. Since constraints (6.9) ensure that no processing units are eliminated or created, no resource bounds on these variables are required.

Constraints (6.11) are needed to determine the materials availability R_{mt}^{DS}. The set material resources M includes raw materials, intermediaries and products, $M = W \cup I \cup P$, of both campaign and short-term products. The coefficient $\nu_{km\theta}$ defines the proportion of materials consumed and produced of the batch size ξ_{kt}^{DS}. The continuous variables Π_{mt}^{DS} express the deliveries of the products at each the time interval t. We assume that Π_{mt}^{DS} will always have non positive values, thus no material receipts are expected to occur during the scheduling horizon H^{DS}. Constraints (6.12) define the minimum and maximum materials availability allowed for each time interval. Constraints (6.13) ensure that the batch size ξ_{kt}^{DS} is between the minimum V_{ke}^{min} and maximum V_{ke}^{max} allowed capacities of the processing units e and are just defined for the non-regular products, while constraints (6.14) define the minimum and maximum lot size of the campaign tasks (regular products).

On the demand side, the variables Π_{mt}^{DS} must be equal to zero for all materials, except for final products, see constraints (6.15), and at the time intervals different of the delivery dates t_d, see constraints (6.16). The minimum and maximum amount of each delivery is specified by constraints (6.17). Production requirements were modeled as "soft constraints" to avoid schedule infeasibilities. The missing deliveries are expressed by the continuous variables $\Pi_{md}^{DSslack}$, which are penalized in the objective function through coefficient α_m. Practice demonstrates that this is often the case when dealing with medium and long term scheduling. Finally, expressions (6.18) express the variables domain.

Constraints

$$R_{pet}^{DS} = \left(R_{pe}^{init}|_{(t=0)}, R_{pe,t-1}^{DS}|_{t>0}\right) + \sum_{\substack{(p',k)\in KP_e \\ p'=p}} \sum_{\theta=0}^{\tau_k} \mu_{ke\theta} N_{k,t-\theta}^{DS} + \sum_{p'\in P_e} \sum_{p''\in P_e} \sum_{\theta=0}^{c_{ep'p''}} \alpha_{ep'p''p\theta} C_{ep'p'',t-\theta}^{DS} \quad \forall p \in P, e \in E, t \in H^{DS} \tag{6.9}$$

$$\sum_{p\in P_e} R_{pe}^{init} \le R_e^{init} \quad \forall e \in E \tag{6.10}$$

$$R_{mt}^{DS} = \left(R_m^{init}|_{(t=0)}, R_{m,t-1}^{DS}|_{(t>0)}\right) + \sum_{k\in K_m} \sum_{\theta=0}^{\tau_k} \nu_{km\theta} \xi_{k,t-\theta}^{DS} + \Pi_{mt}^{DS} \;\; \forall m \in M, t \in H^{DS} \tag{6.11}$$

$$0 \le R_{mt}^{DS} \le R_{mt}^{max} \quad \forall r \in M, t \in H \tag{6.12}$$

$$V_{ke}^{min} N_{kt}^{DS} \le \xi_{kt}^{DS} \le V_{ke}^{max} N_{kt}^{DS} \quad \forall e \in E, k \in K_e^{NR}, t \in H^{DS} \tag{6.13}$$

$$L_{rk}^{min} N_{kt}^{DS} \le \xi_{kt}^{DS} \le L_{rk}^{max} N_{kt}^{DS} \quad \forall r \in P, k \in K^{R}, t \in H^{DS} \tag{6.14}$$

$$\Pi_{mt}^{DS} = 0 \;\; \forall m \in M\backslash P, t \in H^{DS} \tag{6.15}$$

$$\Pi_{mt}^{DS} = 0 \quad \forall m \in P, t \in H^{DS}\backslash\{t_d\}_{d\in D_m^{DS}} \tag{6.16}$$

$$Q_{md}^{DSmax} \ge -\Pi_{m,t_d}^{DS} \ge Q_{md}^{DSmin} - \Pi_{md}^{DSslack} \;\; \forall m \in P, d \in D_m^{DS} \tag{6.17}$$

$$\begin{gathered} R_{pet} \in \mathbb{R}_+ \quad \forall\, \mathrm{p} \in \mathrm{P}, e \in E, t \in H^{DS} \\ R_{mt}^{DS} \quad \forall m \in M, t \in H^{DS} \\ \xi_{kt} \in \mathbb{R}_+ \quad \forall k \in K, t \in H^{DS} \\ \Pi_{md}^{slack} \in \mathbb{R}_+ \;\; \forall m \in P, d \in D_m^{DS} \\ \Pi_{mt}^{DS} \in \mathbb{R}_- \quad \forall m \in M, t \in H^{DS} \\ N_{kt} \in \{0,1\} \quad \forall\, k \in K, t \in H \\ C_{ep'p''t} \in \{0,1\} \quad \forall e \in E, p', p'' \in P_e, t \in H \end{gathered} \tag{6.18}$$

Objective Function

The objective function is given by expression (6.19) and maximizes the profit, taking into account the value of the products, inventory costs of the materials and changeovers costs. The last term introduces a penalty cost for missing deliveries.

$$\max\left[\sum_{m\in P}\sum_{d\in D_m^{DS}} -\Pi_{\mathrm{m,t_d}}^{DS}(v_m - c_m^{raw}) - \sum_{m\in M}\sum_{t\in H^{DS}} c_m^{sto} R_{mt}^{DS} - \sum_{e\in E}\sum_{p\in P}\sum_{p'\in P}\sum_{t\in H^{DS}} c_e C_{epp't} - \sum_{m\in P}\sum_{d\in D_m^{DS}} \alpha_m \Pi_{md}^{DSslack}\right] \tag{6.19}$$

6.6 Solution Methods

The DS model can directly be solved using an exact method such as the branch-and-bound (B&B). However, with the increase of the number of resources or the number of tasks or, mainly, with the increase of the time periods (resulting from decreasing the duration of the time intervals or increasing the scheduling horizon), the model would lead to large optimization problems that would hardly be solved by exact methods in acceptable amount of time. Alternatively, decomposition approaches can be applied to obtain satisfactory solutions quickly.

In this work, we have decided to apply a rolling horizon approach based on the works by Dimitriadis et al. (1997) and Erdirik-Dogan and Grossmann (2007), and the reformulation and branching strategy proposed by Velez and Maravelias (2013). The rolling horizon approach considers the detailed scheduling model (DS) and an aggregate planning model (AP), and is applied as depicted in Figure 6.6. The algorithm will progressively increase the horizon of the DS model and shrink the horizon of the AP model. The reformulation and branching strategy goal is to improve the performance of the B&B by reducing the symmetry of the scheduling solutions. Several modifications were performed in both methods so as to improve their performance.

6.6.1 Aggregate Planning Model (AP model)

The main objectives of the AP model are to obtain a fair estimative of the scheduling solution, at a low computational time, and to trigger adequate production needs at the time interval boundary with the DS model, when running the rolling horizon approach. To achieve that, sequencing and detailed timing variables of the DS model were ignored and the planning horizon was divided in periods having duration of one week. Product demand and the corresponding deliveries take place only at these periods; therefore they are called delivery periods. Note that, the AP model considers the same delivery periods as the DS model.

Although solutions obtained by the AP model cannot be applied because tasks-sequencing are not modeled, the model yields upper bounds on the profit value. The AP model is based on the aggregate planning model proposed by Erdirik-Dogan and Grossmann (2007) and is defined by constraints (6.20) to (6.30) and objective function (6.31). We have considered the continuous variables R_{md}^{AP} that define the availability of material m at delivery period d, the continuous variables ξ_{kd}^{AP} that define the total amount of material processed by task k at delivery period d and the continuous variables Π_{md}^{AP} that define the amount delivered of final product m at delivery period d.

Materials balance constraints (6.20) are defined for all delivery periods and materials. The proportion of material consumed and produced is given by the parameter ν_{km}. Since the detailed timing and sequencing constraints have not been considered, there is no need to model the availability of the processing units. The minimum and maximum materials availability is given by constraints (6.21); the demand “soft-constraints” are given in (6.22); and deliveries cannot take place for raw materials and intermediaries, see constraints (6.23) and (6.24). Constraints (6.25) and (6.26) bound the total amount of material processed by tasks of the non-regular products and of the campaign tasks respectively. They are similar to constraints (6.13) and (6.14) of the DS model, however in the AP model they are required to compute the number of batches of each task (the integer variables N_{kd}^{AP}).

The production capacity is expressed in terms of time available in the processing units by delivery period it_d^{AP}. The first summation of constraints (6.27) defines the total time required by tasks $k \in K_e$ in processing unit e and the second summation accounts for an estimation of the changeovers times. The binary variables Y_{ped}^{AP} determine if product p is produced in unit e at delivery period d and the parameter $chg^{AP}$is the changeover time, which is assumed to be equal to all products and units. Since the tasks sequence is not known, the expression $\sum_{p\in P}(Y_{ped}^{AP}\, chg^{AP})$ could lead to an overestimation of the changeovers times in the cases that the unit ends with one product in delivery period d and starts with the same product in delivery period $d+1$. Thus, the third term of constraints (6.27) is added so as to express the fact that the number of changeovers is equal to the number of products minus one. Constraints (6.28) and (6.29) are used to define the variables Y_{ped} and constraints (6.30) to define the variables domain.

The task processing times is given by the parameter τ_{ke}, but is defined in a different way for the regular and non-regular products. Since the regular products are modeled through campaign tasks, τ_{ke} value is equal to the sum of the processing times of all tasks assigned to unit *e* in the campaign task. Thus in campaign tasks, τ_{ke} retrieves the total time campaign task k requires from processing unit e. Regarding the non-regular products, the value of τ_{ke} is just determined by the processing time of each task, so $\tau_{ke}=\tau_k$.

Constraints

$$R_{md}^{AP} = \left(R_m^{APinit}|_{d=0}, R_{m,d-1}^{AP}|_{d>0}\right) + \sum_{k\in K_m} \nu_{km}\xi_{kd}^{AP} + \Pi_{md}^{AP} \quad \forall m\in M, d\in D^{AP} \tag{6.20}$$

$$0 \le R_{md}^{AP} \le R_{md}^{max} \quad \forall m\in M, d\in D^{AP} \tag{6.21}$$

$$Q_{md}^{DSmax} \ge -\Pi_{md}^{AP} \ge Q_{md}^{DSmin} - \Pi_{md}^{APslack} \;\; \forall m\in P, d\in D^{AP} \tag{6.22}$$

$$\Pi_{md}^{AP} = 0 \;\; \forall m\in M\backslash P, d\in D^{AP} \tag{6.23}$$

$$\Pi_{md}^{APslack} = 0 \;\; \forall m\in M\backslash P, d\in D^{AP} \tag{6.24}$$

$$V_{ke}^{min} N_{kd}^{AP} \le \xi_{kd}^{AP} \le V_{ke}^{max} N_{kd}^{AP} \quad \forall e\in E, k\in K_e^{NR}, d\in D^{AP} \tag{6.25}$$

$$L_k^{min} N_{kd}^{AP} \le \xi_{kd}^{DS} \le L_k^{max} N_{kd}^{AP} \quad \forall k\in K^R, d\in D^{AP} \tag{6.26}$$

$$\sum_{k\in K_e} N_{kd}^{AP}\tau_{ke} + \sum_{p\in P} Y_{ped}^{AP}\, chg^{AP} - chg^{AP} \le it_d^{AP} \quad \forall e\in E, d\in D^{AP} \tag{6.27}$$

$$N_{kd}^{AP} \geq Y_{ped}^{AP} \quad \forall e \in E, (p,k) \in KP_e, d \in D^{AP} \tag{6.28}$$

$$N_{kd}^{AP} \leq \left\lfloor \frac{it_d^{AP}}{\tau_{ke}} \right\rfloor Y_{ped}^{AP} \quad \forall e \in E, (p,k) \in KP_e, d \in D^{AP} \tag{6.29}$$

$$\begin{gathered} R_{md}^{AP} \in \mathbb{R}_+ \quad \forall m \in M, d \in D^{AP} \\ \xi_{kd}^{AP} \in \mathbb{R}_+ \quad \forall k \in K, d \in D^{AP} \\ \Pi_{md}^{slack} \in \mathbb{R}_+ \ \forall m \in P, d \in D \\ \Pi_{md}^{AP} \in \mathbb{R}_- \quad \forall m \in M, d \in D^{AP} \\ N_{kd}^{AP} \in \mathbb{Z}_+ \quad \forall k \in K, d \in D^{AP} \\ Y_{ped}^{AP} \in \{0,1\} \quad \forall p \in P, e \in E, d \in D^{AP} \end{gathered} \tag{6.30}$$

Objective Function

The objective function (6.31) aims at maximizing the profit and is similar to the objective function of the DS model, differing only in the time and tasks sequencing aggregation.

$$\begin{aligned} \max \Bigg[&\sum_{m \in P} \sum_{d \in D_m^{AP}} -\Pi_{md}^{AP}(v_m - c_m^{raw}) \\ &- \sum_{m \in M} \sum_{d \in D^{AP}} c_m^{sto} R_{md}^{AP} \\ &- \sum_{e \cup E} \sum_{p \in P} \sum_{d \in D^{AP}} c_e Y_{ped} - \sum_{m \in P} \sum_{d \in D_m^{AP}} \alpha_p \Pi_{md}^{DSslack} \Bigg] \end{aligned} \tag{6.31}$$

6.6.2 Rolling Horizon (RH Approach)

The RH approach is defined by the DS model constraints for the detailed scheduling horizon and by the AP model constraints for the aggregate planning horizon. The objective is to maximize the profit given by the sum of the objective functions of the two models.

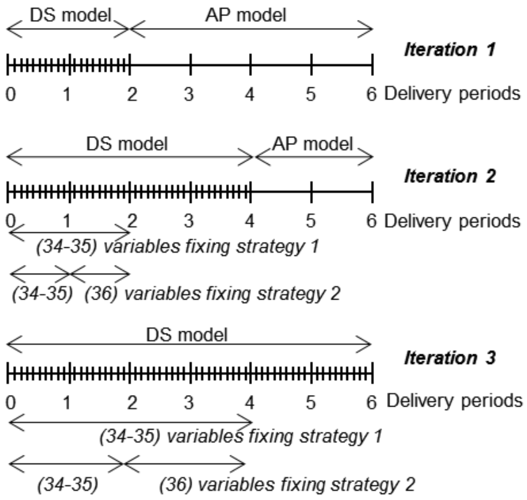


Figure 6.6 – Rolling horizon approach.

Figure 6.6 depicts three iterations of the RH approach considering a scheduling horizon of six weeks and a rolling horizon window of two weeks. In each iteration, the task-unit assignment binary variables N_{kt}^{DS} determined in the previous iteration are fixed. By fixing those variables the computational complexity of the DS is reduced while some flexibility is kept on the batch size continuous variables ξ_{kt}^{DS}. In the last iteration, the DS model is applied to the entire scheduling horizon. Two different fixing strategies are tested. This will be explained in detail below.

An important choice in this approach is the length of the scheduling horizon (rolling horizon window) that the DS model should consider. This length cannot be too large as it would result in prohibitive solution times of the DS model, and it cannot be too small as it is limited by the production lead time of the products.

An additional set of constraints is added to link both models. Constraints (6.32) impose that the materials available at the end of the detailed scheduling horizon are equal to the initial amount of materials available for the AP model. Constraints (6.33) enforce that no task is executed in the DS model if it cannot be finished. These constraints are important to ensure feasibility in the intervals boundaries between the DS and AP models, by blocking the occurrence of tasks that may lead to overproduction, as explained by Dimitriadis et al. (1997).

Constraints

$$R_{mT^{DS}}^{DS} = R_m^{APinit} \ \ \forall m \in M \tag{6.32}$$

$$\sum_{t=T^{DS}-\tau_k}^{T^{DS}-1} N_{kt}^{DS} = 0 \ \ \forall k \in K \tag{6.33}$$

$$N_{kt}^{DS} = 1 \ \ \forall k \in K^{fixed}, t \in H^{DSfixed} \tag{6.34}$$

$$C_{epp't}^{DS} = 1 \ \ \forall e \in E^{fixed}, p, p' \in P^{fixed}, t \in H^{DSfixed} \tag{6.35}$$

$$\sum_{k \in A_{k'}} \sum_{t \in H^{DSfixed}} N_{kt}^{DS} \geq NT_{k'}^{fixed} \ \ \forall\, k' \in AK \tag{6.36}$$

Variables Fixing Strategies

As mentioned, two distinct strategies are followed regarding variables fixing (see Figure 6.6). *Strategy 1* is similar to the approach followed by Dimitriadis et al. (1997) and Erdirik-Dogan and Grossmann (2007). Here, the binary variables N_{kt}^{DS} and $C_{epp't}^{DS}$ that are equal to 1 in each iteration of the RH are fixed in the next iteration through constraints (6.34) and (6.35). Additionally, we proposed a mixed approach, *Strategy 2*, which determines, in each iteration of the RH, the number of times a task runs NT_k^{fixed} in the DS model. In the right-hand side of expression (6.36) the parameter NT_k^{fixed} gives the number of tasks occurrences grouped by alternative tasks. The set AK gives the group of tasks, while set $A_{k'}$ gives the tasks considering the existing alternative processing units to task k'. Then, in the following iteration of the RH, constraints (6.34) and (6.35) are applied in the first time intervals, while constraints (6.36) are applied in the last time intervals of the DS model, as shown in Figure 6.6. In this way, *Strategy 2* fixes the binary variables in beginning of the scheduling horizon H^{DS} and allows for some flexibility on the task-unit assignment at the end of this time horizon, where there is the link with the AP model.

6.6.3 Reformulation and Branching Strategies

Velez and Maravelias (2013) studied the scheduling problem and proposed a reformulation for the MILP model that considers new integer variables NT_k^{DS} for determining the number of times task k runs. The authors demonstrated that giving higher branching priority to the NT_k^{DS} variables lead to the elimination of many symmetric solutions and improved the computational performance of the scheduling model. To account for this approach, constraints (6.37) and (6.38) are added to the DS model. Moreover, since the DS model accounts for sequence-dependent changeovers, we propose new integer variables NC_p^{DS} to determine the number of changeovers associated to product p. These new variables are defined by constraints (6.39).

Constraints

$$\sum_{t \in H^{DS}} N_{kt}^{DS} = NT_k^{DS} \quad \forall k \in K \tag{6.37}$$

$$0 \leq NT_k^{DS} \leq \left\lfloor \frac{T^{DS}}{\tau_k} \right\rfloor \quad \forall k \in K \tag{6.38}$$

$$\sum_{e \in E} \sum_{p' \in P} \sum_{p'' \in P} \sum_{t \in H^{DS}} C_{ep'p''t}^{DS} = NC_p^{DS} \;\; \forall p \in P: (p' = p \vee p'' = p\,) \tag{6.39}$$

6.7 Results

In this section, we propose to solve the illustrative example depicted in Figure 6.7 and a real case-study from a chemical-pharmaceutical industry shown in Figure 6.1. The proposed algorithm for regular and non-regular production scheduling and the solution methods are tested for several time horizons (4, 8 and 12 weeks). Although scheduling scenarios using campaign tasks cannot be directly compared with scenarios that consider the detailed recipes, since they target different scheduling solutions, we extensively compare both scenarios so as to evaluate the impact of the cyclic operation in the schedules.

The formulations used are summarized in Table 6.2 and were implemented using ILOG/CPLEX version 12.5.1, running on an Intel Xeon at 3.33GHz machine with 24 GB of RAM. We test three reformulations of the DS model. DS1 and DSp1 models account for the reformulation and branching priority as proposed by Velez and Maravelias (2013).

DS2 model includes the reformulation with the NT_k^{DS} and NC_p^{DS} integer variables and giving no branching priority. Finally, RH1 implements the variables fixing strategy 1 and RH2 considers the variables fixing strategy 2. In strategy 2, constraints (6.34) and (6.35) are applied to the detailed scheduling horizon minus the last week, while constraints (6.36) are applied over the last week. For example, in the 8 weeks scheduling and assuming a rolling horizon window of 3 weeks, in the second iteration of the RH constrains (6.34) and (6.35) are applied in the two first weeks and constrains (6.36) are applied in the third week.

Table 6.2 – Formulations.

Model	Description	Formulation
AP	Aggregate planning model	(6.20) to (6.31)
DS	Detailed scheduling model	(6.9) to (6.19)
DS1	Detailed scheduling model with reformulation 1	(6.9) to (6.19), (6.37) and (6.38)
DSp1	Detailed scheduling model with reformulation 1 and branching priority on the variables NT_k^{DS}	(6.9) to (6.19), (6.37) and (6.38)
DS2	Detailed scheduling model with reformulation 2	(6.9) to (6.19) and (6.37) to (6.39)
RH1	Rolling horizon with variables fixing strategy 1	(6.9) to (6.18), (6.20) to (6.30) and (6.32) to (6.35)
RH2	Rolling horizon with variables fixing strategy 2	(6.9) to (6.18), (6.20) to (6.30) and (6.32) to (6.36)

6.7.1 Illustrative Example

Here we solve a scheduling problem of reduced size, where 3 products requiring each, one reaction task and one filtering task, are considered. The reaction tasks take 16 hours and can only be executed by reactor U1, while filtering tasks take 8 hours and have two suitable filters, F1 and F2 (see Figure 6.7).

Since the recipes of the products have a similar structure, they can be represented by similar campaign tasks, as depicted in Figure 6.8. The maximum capacity of unit U1 is 5 tons, and of the filters is 3 tons. Raw materials and final products have finite intermediate storage (FIS) and intermediaries follow a zero-wait storage policy (ZW). Products P1, P2 and P3 economic values are 10, 20 and 15 monetary units (m.u.); the raw material costs are 5, 3 and 6 m.u.; and the storage costs 0.05, 0.08 and 0.04 respectively. Since all the processing times of the tasks are multiple of 8, the time periods were assumed to have a fixed duration of 8 hours. The sequence-dependent changeover is of 24 hours and equal to all three products. The missing delivery costs α_m are twice the value of the products.

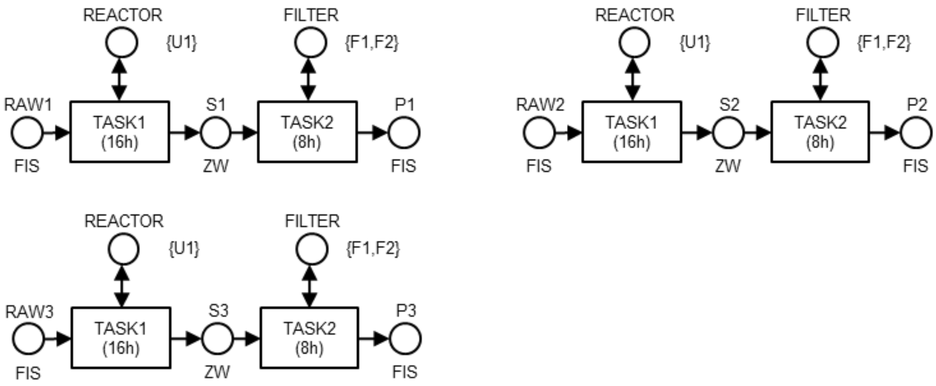


Figure 6.7 – Product recipes for the illustrative example.

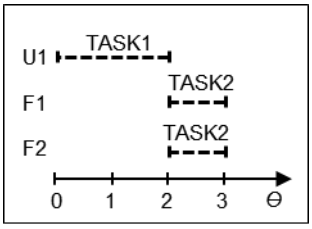


Figure 6.8 – Illustrative example: campaign task structure.

The numerical results shown in tables present the following data: the model used; the scheduling time horizon; the number of integer and continuous variables and constraints; the number of nodes and iterations; the value of the linear relaxation of the MILP; the integrality gap; the objective function value and the computational time required for solving the instance. Regarding the RH approach the data shown is related to the last iteration, with exception of the CPU time column that displays the total time required by the algorithm.

6.7.1.1 4 Weeks Scheduling

The solution statistics of the four weeks scheduling problem are given in Table 6.3. As can be seen, all models with the exception of the AP model obtained the optimal solution of 1,962.3 m.u.. The RH1 required less CPU time than the other models, obtaining the optimal solution in just 11.9 CPU seconds (assuming that no campaign tasks are used). Note that, RH1 and RH2 have rolling horizon windows equal to 2 delivery periods. Moreover, results show that DSp1 model required more than twice the CPU time of DS, DS1 and DS2 models, having also higher number of nodes and iterations.

Using campaign tasks for the three products, the instance size reduced as well as the CPU time needed to solve the problem (see Table 6.10 in Appendix B). Again the RH1 had the best performance, obtaining the optimal solution in just 5.4 CPU seconds.

Table 6.3 – Four weeks schedule (4W = four weeks scheduling horizon).

Model/horizon	Int. variables/cont. variables/constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
AP/4W	72/145/300	117	673	2,397.36	0.01	2,119.8	0.7
DS/4W	2295/3337/5646	12,259	2,183,279	2,245.55	0.01	1,962.3	64.5
DS1/4W	2304/3337/5664	16,731	2,292,919	2,245.55	0.01	1,962.3	60.6
DSp1/4W	2304/3337/5664	26,301	5,880,359	2,245.55	0.01	1,962.3	142.1
DS2/4W	2307/3337/5667	13,211	1,750,774	2,245.55	0.00	1,962.3	50.9
RH1/4W	2295/3337/5703	16	3,787	2,011.01	0.00	1,962.3	11.9
RH2/4W/	2295/3337/5650	10,225	1,261,106	2,192.70	0.00	1,962.3	30.0

6.7.1.2 8 Weeks Scheduling

With the increase of the scheduling horizon to eight weeks and assuming no campaign tasks, none of the DS models proved optimality in the time limit of 3600 CPU seconds (see Table 6.4). DS2 requiring just three more binary variables and constraints than DS1, performs better computationally.

Assuming campaign tasks the instance became easier to solve and DS, DS1 and DS2 models proved optimality within the 3600 CPU seconds. Additionally, results show that the reformulation DS2 had better performance than the DS, DS1 and DSp1 models, and that it seems preferable to use the default CPLEX branching priority, instead of giving priority to the NT_k^{DS} variables. In both instances DSp1 had the worst performance.

Table 6.4 – Eight weeks schedule (8W = eight weeks scheduling horizon; C = campaign tasks used).

Model/horizon/aggregation	Int. variables/cont. variables/constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
AP/8W	144/289/600	3,537	27,943	4,967.20	0.01	4,553.9	1.0
DS/8W	4563/6625/11214	245,719	77,233,056	4,631.96	2.78	4,111.5	3,601.4
DS1/8W	4572/6625/11232	323,250	101,340,287	4,631.96	3.09	4,104.0	3,601.5
DSp1/8W	4572/6625/11232	196,492	88,934,784	4,631.96	8.63	3,940.7	3,601.3
DS2/8W	4575/6625/11235	361,553	109,406,063	4,631.96	1.79	4,117.5	3,601.5
DS/8W/C	3549/4597/7152	1,027,859	115,570,751	4,630.48	0.01	4,121.5	2,165.0
DS1/8W/C	3552/4597/7158	755,217	64,563,162	4,630.48	0.01	4,121.5	1,133.3
DSp1/8W/C	3552/4597/7158	694,648	160,640,734	4,630.48	2.01	4,121.5	3,601.3
DS2/8W/C	3555/4597/7161	639,363	56,872,981	4,630.48	0.01	4,121.5	956.2

In Figure 6.9, it is represented the CPU times and objective function values assuming that recipes are aggregated using campaign tasks. Concerning just the computational time, the RH is certainly the most competitive method. While DS2 required 956.2 CPU seconds to obtain a solution of 4,121.5 m.u., RH2.2 (rolling horizon window equal to 4 delivery periods) just took 63.7 CPU seconds to obtain a solution with profit equal to 4,112.2 m.u. (see Table 6.11 and Table 6.12 in Appendix B). Nevertheless, results indicate that RH approach is very dependent on the definition of the variables fixing strategy and the length of the rolling horizon window. As shown in Figure 6.9, for the same rolling horizon window the RH2 had always better results than RH1 and solutions tend to improve with the increase of the rolling horizon window. The lower profit solution was obtained by RH1/8W/C which is by 5% less than the best solution found. The best solution among the RH methods was retrieved by RH2.2/8W/C that assumes a rolling horizon window of 4 delivery periods and is just by 0.2% inferior to the best solution found.

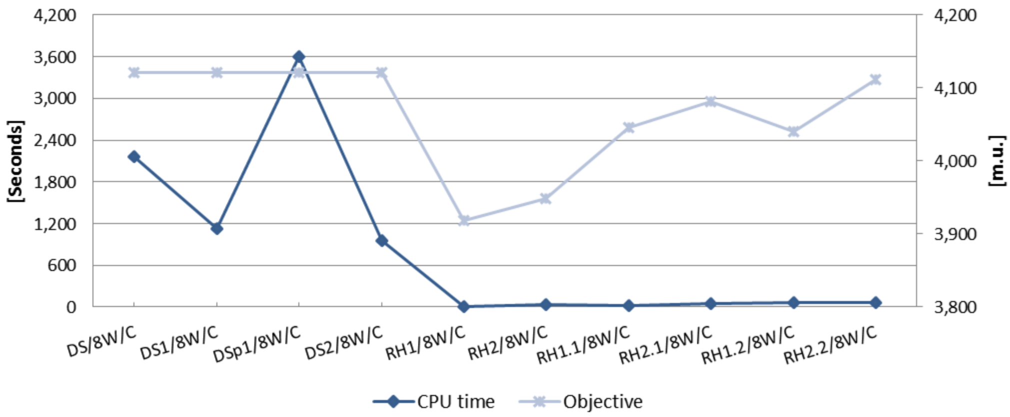


Figure 6.9 – Eight weeks schedule: models computational times and objective function values. (RH1 and RH2 have rolling horizon windows equal to 2 delivery periods; RH1.1 and RH2.1 have rolling horizon windows equal to 3 delivery periods; RH1.2 and RH2.2 have rolling horizon windows equal to 4 delivery periods).

6.7.1.3 12 Weeks Scheduling

The longest scheduling horizon this paper considers is of 12 weeks. The best solution found was obtained by the DS model in 3,395 CPU seconds, assuming campaign tasks, with a profit of 6,182.8 m.u. (see

Figure 6.10). The lower profit solution, among the DS models, was retrieved by DSp1. Although the use of campaign tasks leads to a reduction of more than 20% of the number of integer variables, the DS1 and DS2 models could not deliver solutions within an integrality gap of 5% and time limit of 3600 CPU seconds.

Once more the quality of the solutions delivered by the RH approaches strongly depends on the variables fixing strategy and on the length of the rolling horizon window. The lower profit solution was retrieved by RH2 and is by 6 % inferior to the best solution. The best solution among the RH approaches was obtained in just 132.9 CPU seconds by RH2.2 with a profit of 6,162.1 m.u..

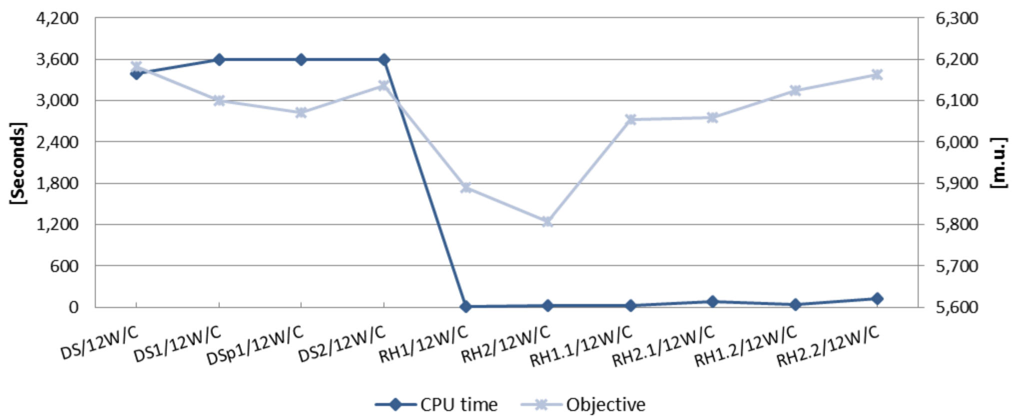


Figure 6.10 – Twelve weeks schedule: models computational times and objective function values. (RH1 and RH2 have rolling horizon windows equal to 2 delivery periods; RH1.1 and RH2.1 have rolling horizon windows equal to 3 delivery periods; RH2.2 and RH2.2 have rolling horizon windows equal to 4 delivery periods).

Overall, the DS models worked reasonably well. However, with the increase of the scheduling horizon the DS models could not prove optimality. In opposition, the AP model retrieved solutions in very short times but overestimated the production capacity. Looking into the 4, 8 and 12 weeks problems, we can conclude that using campaign tasks improves the computational performance of the models. The RH approaches ran quite fast and obtained good solutions or even optimal solutions. The variables fixing strategy and the length of the rolling horizon window strongly affect the quality of the solutions. In general, the variables fixing strategy 2 requires more CPU time, but obtains better solutions than strategy 1. This is related to the flexibility of constraints (6.36) that do not impose a fix task-unit assignment for the time intervals of the DS model that interface with the AP model. In other words, task-unit assignment is allowed to change in order to better accommodate the production requirements in the next iteration of the RH approach, while the CPU time required to solve the DS model is kept low. We can expect better solutions if a larger RH window is considered, since scheduling decisions are taken considering more data. Nevertheless, it is important to note that with the increase of the rolling horizon window the scheduling problem becomes more difficult to solve. Therefore, RH window size must be defined taking into account the CPU time required to solve the scheduling problem. In order to emphasize the complexity of modeling sequence-dependent changeovers in scheduling problems, we note that the 12 weeks instance without changeovers can be solved to optimality in less than 1 second (see Table 6.15 in Appendix B).

6.7.2 Real Case Study

In this section, we solve a real-world scheduling problem from a chemical-pharmaceutical industry. We consider a multipurpose batch plant producing the 5 products depicted in Figure 6.1. These are to be scheduled in a time horizon of up to 3 months and the schedule must give the tasks-unit assignment and sequencing of the regular and non-regular products.

Product PA recipe has 11 tasks and one stable intermediary (PA_S4), requiring a production time of 304 hours (sum of the tasks processing times required to manufacture one batch). Product PB has 5 tasks, one stable intermediary and a total production time of 72 hours. Product PC has 6 tasks that require a total of 128 hours. Product PD has 10 tasks and takes 184 hours. Finally, Product PE has 11 tasks and takes 224 hours. The objective is the profit maximization. The scheduling horizon was discretized into time intervals of 8 hours, since all task durations are assumed to be multiples of 8. The sequence-dependent changeover tasks take 24 hours and the missing delivery costs α_m are twice the value of the products.

We have considered two different production types: non-regular and regular production. The products that are produced in a regular basis have been assigned to specific production lines, while the non-regular products have more flexibility regarding the task-unit assignment. Note that, in the course of the process development of a new drug, the set of alternative processing units available for each task tends to become smaller leading to stable and well-defined recipes. Thus, in its operation the company considers Products PA and PB as regular products that are represented here by the respective campaign tasks (see Figure 6.5) and products PC, PD and PE as non-regular products, which are represented by their detailed recipe as depicted in Figure 6.1. The case study is solved considering 4, 8 and 12 weeks scheduling horizons scenarios.

In the DS models we have assumed two stopping criteria, the integrality gap of 5% and time limit of 14,400 seconds, and in the RH approaches, we have considered the integrality gap of 5% and time limit of 3,600 seconds.

6.7.2.1 4 Weeks Scheduling

The results of the 4 weeks scheduling instance are shown in Table 6.5. The best solution, without campaign tasks, was obtained by DS1 and DSp1 models in 14,400 CPU seconds, with a profit equal to 36,684.7 m.u., while DS2 delivered a solution within 4.98% of the optimum in just 7,958.6 CPU seconds. Overall, the RH approaches performed quite well. For example, the solution of RH2 was obtained in just 3,610.4 CPU seconds and is by 2% inferior to the best solution found.

Modeling the regular products PA and PB with campaign tasks led to reduction of the profit by 7% to 34,268.0 m.u. The storage costs are higher when using campaigns since it is required keeping stock of the stable intermediaries. This can be interpreted as the cost of the cyclic operation for the regular products. Additionally, note that campaign tasks impose strict tasks sequencing for the regular products, which results in a loss of flexibility when performing scheduling. On the other hand, campaign tasks allow the definition of production lines with cyclic operation, and the control over the inventory of the stable intermediaries, leading to more responsive schedules. The DS model had the best performance among the detailed models, and the RH2 approach obtained a solution within 6% of the best solution, in just 76.6 CPU seconds. Again, results show that RH2 achieved better results when compared with RH1, but at cost of higher CPU time.

Table 6.5 – Four weeks schedule (4W = four weeks scheduling horizon; C = campaign tasks used).

Model/horizon/aggregation	Int. variables/cont. variables/constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
DS/4W	15045/18937/33525	198,489	180,675,428	39,524.40	5.52	36,441.2	14,402.9
DS1/4W	15110/18937/33655	92,545	94,107,274	39,524.40	5.06	36,684.7	14,414.2
DSp1/4W	15175/18937/33655	91,782	93,376,456	39,524.40	5.06	36,684.7	14,412.6
DS2/4W	15115/18937/33660	61,821	66,295,967	39,524.40	4.98	36,612.5	7,958.6
RH1/4W	15045/18937/33618	8,098	2,337,094	37,268.50	4.99	34,206.9	136.2
RH2/4W	15045/18937/33589	33,768	34,907,653	39,128.90	5.03	36,102.3	3,610.4
DS/4W/C	13855/15707/27051	106,882	156,144,030	37,946.40	6.19	34,268.0	14,402.7
DS1/4W/C	13906/15707/27153	84,583	172,741,014	37,946.40	10.80	33,177.3	14,402.1
DSp1/4W/C	13957/15707/27153	83,771	170,967,461	37,946.40	10.83	33,177.3	14,402.6
DS2/4W/C	13911/15707/27158	70,537	111,335,068	37,946.40	5.00	34,179.9	9,311.8
RH1/4W/C	13855/15707/27079	0	5,244	33,470.90	4.78	31,329.4	22.6
RH2/4W/C	13855/15707/27076	1,859	727,083	36,505.10	3.37	32,275.3	76.6

Figure 6.11 depicts the schedule solution of approach RH2/4W/C, thus assuming that campaign tasks are used to model the regular products PA and PB. As can be seen, two campaigns of PA and four campaigns of PB are scheduled. The first campaign of PB starts in week 1 and runs three campaign cycles. At the end of this week 360 kg of PB are delivered. This campaign is then interrupted to produce one campaign cycle of PA that delivers 235 kg of this product, at the end of week 2. Then, the second campaign of PB starts, having also three cycles and delivering 360 kg of this product in week 2. The third campaign of PB is initiated in week 3 and has three cycles. At the end of this week, 360 kg of PB, 64 kg of PC and 208 kg of PE are delivered. In the last week, the second campaign of PA and the fourth campaign of PB are performed, delivering 235 kg of PA, 360 kg of PB and 200 kg of PE.

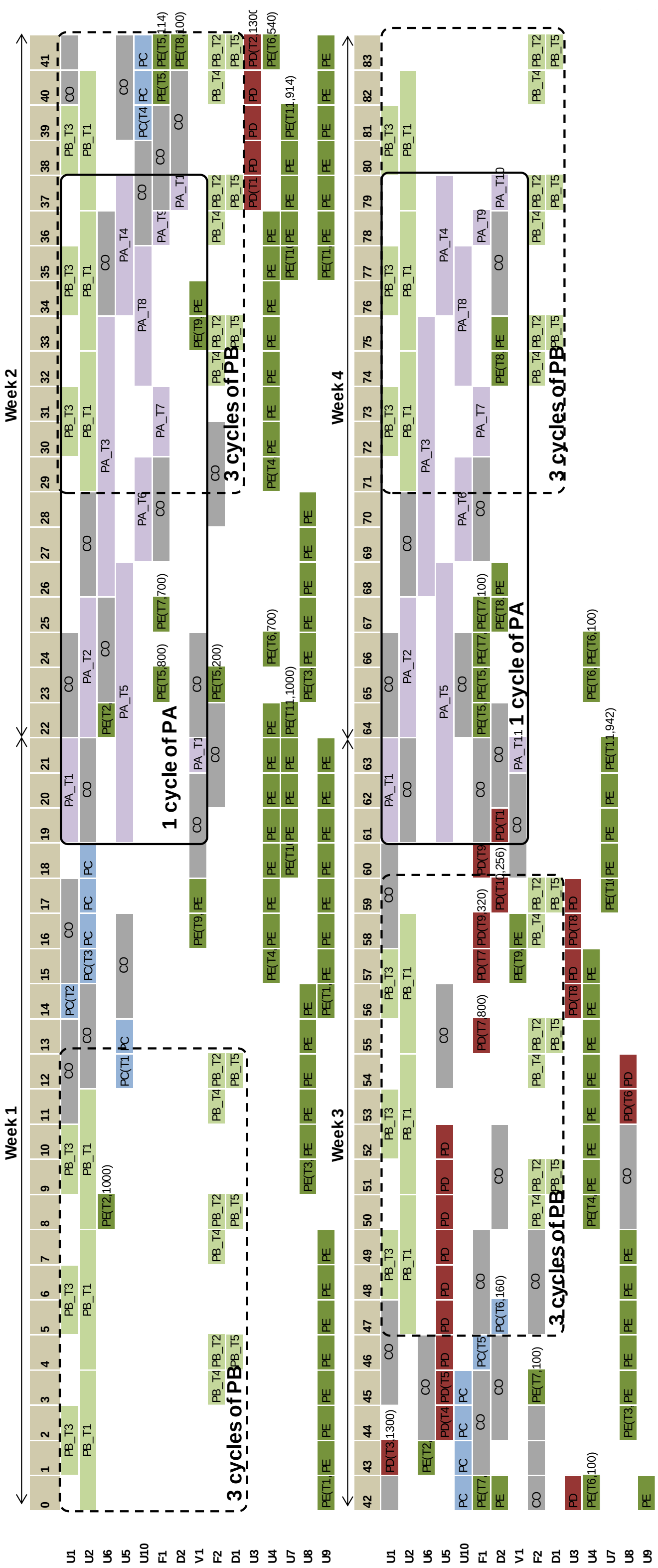


Figure 6.11 – 4 Weeks scheduling of regular and non-regular production (model RH2/4W/C).

6.7.2.2 8 Weeks Scheduling

In the 8 weeks scheduling the RH approaches performed better than the DS models, as can be seen in Figure 6.12. The CPU times of the RH approaches are significantly inferior to the CPU times required by the DS models and the best solutions found in the scenarios with and without campaign tasks were delivered by the RH2 model.

Assuming campaign tasks, RH2 reached a profit of 60,048.9 m.u., which is by 5% inferior to the profit considering that no campaign tasks are used. Among the DS models, the time limit of 14,400 CPU seconds was not sufficient to obtain good quality solutions. The use of the reformulation and branching strategies presented in section 6.6.3 were not advantageous in this instance, since the resultant integrality gaps were higher than 30% (see in Table 6.16 in Appendix B).

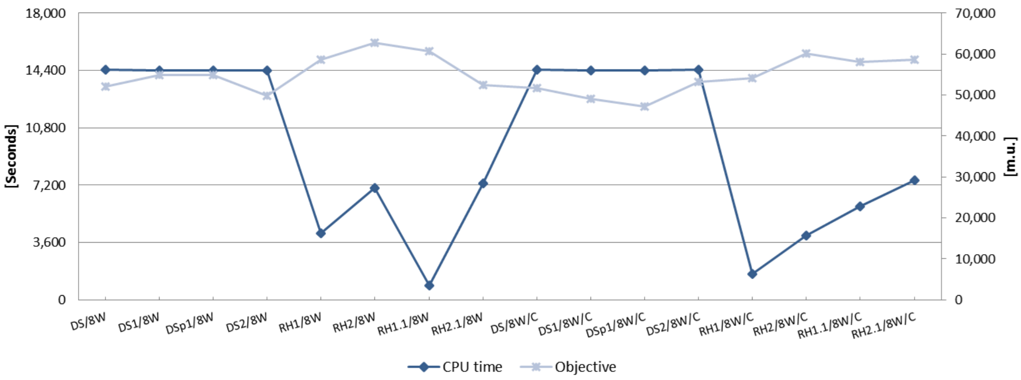


Figure 6.12 – Eight weeks schedule: models computational times and objective function values. (RH1 and RH2 have rolling horizon windows equal to 2 delivery periods; RH1.1 and RH2.1 have rolling horizon windows equal to 3 delivery periods).

6.7.2.3 12 Weeks Scheduling

In the 12 weeks instance, we opted to just apply the RH approach (see Figure 6.13), since in the 8 weeks scheduling horizon the DS models demonstrated to be computationally intractable.

Without campaign tasks, the best solution found has a profit of 91,909.2 m.u. and was obtained by RH2.1 in 5,491.0 CPU seconds. Assuming campaign tasks RH2.1 obtained as well the best solution with a profit of 83,801.40 m.u., which is by 9% inferior to the scenario that does not consider campaign tasks. The RH approaches demonstrated to be a good alternative when exact methods (as are the DS models presented in this paper) tend to obtain solutions with high integrality gaps. In practical terms, the CPU time required by the RH approaches to solve the 3 months scheduling problem has been considered acceptable by the company.

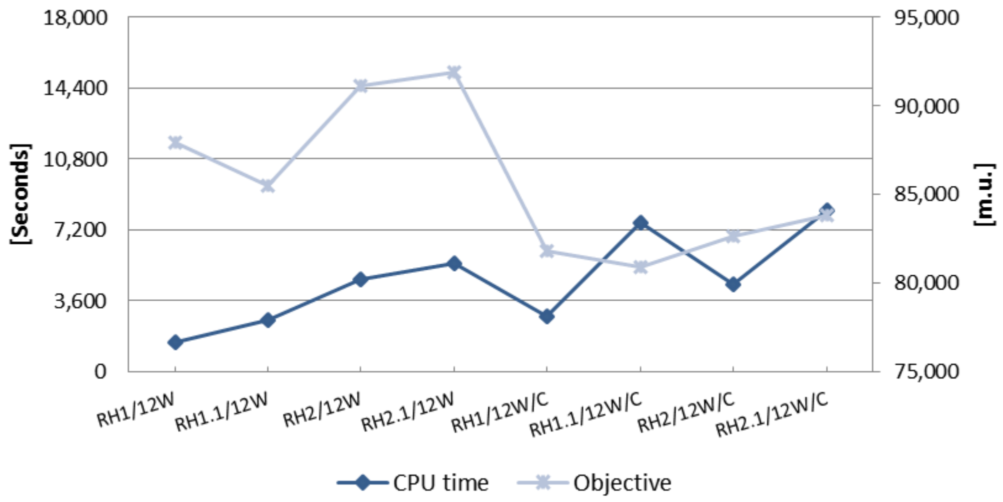


Figure 6.13 – Twelve weeks schedule: models computational times and objective function values. (RH1 and RH2 have rolling horizon windows equal to 2 delivery periods; RH1.1 and RH2.1 have rolling horizon windows equal to 3 delivery periods).

Generally, the definition of campaign tasks responds to one important requirement that we have found in the chemical-pharmaceutical industry: products with well-defined recipes are typically produced in the same processing units and follow predefined production sequences. Moreover, the number of binary and continuous variables and constraints decreased as a result of the task and resource aggregation done in the campaign tasks. Campaign tasks provide more responsive schedules by decreasing the lead time, but may have higher storage costs as a result of the storage policy for the stable intermediaries. The definition of these aggregate tasks allows as well a variation of the amounts being produced, limited by a minimum and maximum production lot, which is not possible to achieve if the typical periodic scheduling approach is applied.

6.8 Conclusions

This paper addresses the scheduling multipurpose of batch plants that simultaneously consider two different operating conditions – regular and non-regular production. The former encompasses the products that are manufactured regularly in predefined production lines and the latter includes under development products having no defined production lines.

A solution approach to solve such problem is proposed, which was developed along mathematical formulations based on RTN. The approach considers the integration of campaign and short-term scheduling in multipurpose batch plants, and proposes a three-step procedure that firstly determines the campaign schedule, secondly creates the campaign tasks and thirdly obtains a detailed schedule for the campaign and non-regular products. Campaigns are modeled as aggregate tasks that take into account the production resources determined previously, while the non-regular products are modeled using their detailed recipe. Campaign tasks proved to be an efficient concept in the cases where the definition of production lines requires cyclic operation mode, which is the procedure followed at the pilot company of this study. In the case study, the use of campaign tasks led to a reduction of the profit by 7%, 5% and 9% in the 4, 8 and 12 weeks schedules respectively, when compared with the scenarios that do not consider campaign tasks. This profit reduction can be interpreted as the cost of the cyclic operation for the regular products.

To deal with the computational complexity of the larger instances, we have decided to compare the performances of a rolling horizon approach based on Dimitriadis et al. (1997) and Erdirik-Dogan and Grossmann (2007) with the reformulation and branching strategy proposed by Velez and Maravelias (2013). Moreover, we have performed several modifications in both methods in order to improve their performance. We propose a reformulation that considers new integer variables for the number of changeovers. Overall, the reformulation proposed by Velez and Maravelias (2013) together with the proposed reformulation improved the results of the base formulation. The combination of the two reformulations demonstrated better performance when compared with the reformulation of Velez and Maravelias (2013). Nevertheless, numerical results show that it is preferable to use the default CPLEX branching priority.

In the smaller instances, the DS models obtained the best solutions in very competitive time. Increasing the size of the scheduling problem, the DS models led to solutions with high integrality gaps (over than 30%) and required considerable CPU time, while the RH approaches obtained better solutions in very small CPU time. The performance of the RH approaches can be truly improved by adapting the variables fixing strategy and the length of the rolling horizon window to the problem. Additionally, it is important to note that the RH can naturally integrate the reformulation strategies for improving the performance of the algorithm.

For further study the authors aim to address other task-unit and temporal decomposition approaches inspired by current industrial practices. Moreover, improvements on the solutions obtained by the rolling horizon, while keeping this approach tractable for large instances, will be also explored.

Notation

Indices

l	campaign
d	delivery period
e	processing unit
k, k'	task
m	material
p	product
r	resource
t	time interval

Sets

AK	production tasks (without considering processing units)
A_k	alternative tasks to *k*
D_m^{DS}	delivery periods of product *m* of the detailed scheduling model
D^{AP}	delivery periods of the aggregate planning model
D_m^{AP}	delivery periods of product *m* of the aggregate planning model
E	processing units (equipments)
E^{fixed}	processing units that are fixed in the rolling horizon approach
H, H^{DS}	scheduling horizon

$H^{DSfixed}$	time horizon corresponding to the fixed tasks
I	intermediaries
I_l	intermediaries associated to campaign l
L	campaigns
M	materials (raw materials, intermediaries and final products)
K	tasks
K^{fixed}	tasks that are fixed in the rolling horizon approach
K_r	tasks that require resource *r*
K_m	tasks that consume or produce material *m*
K_e	tasks associated to unit *e*
K_l	tasks *k* associated to campaign *l*
K_e^{NR}	tasks of the non-regular products that require unit *e*
K^R	campaign tasks of the regular products
K_e^l	tasks *k* associated to campaign *l* and unit *e*
KP_e	tasks *k* of product *p* associated to unit *e*
P	products
P^{fixed}	products that are fixed in the rolling horizon approach
P_l	products associated to campaign l
P_e	products *p* that can be produced in unit *e*
R	production resources (processing units, intermediaries and final products)
R_l	production resources (processing units, intermediaries and final products) associated to campaign l

Parameters

τ_k	processing time of task *k*
τ_{ke}	processing time of task *k* in unit *e* (used in regular and non-regular products)
$\mu_{kr\theta}$	allocation/release coefficient of resource *r* in task *k* at time θ relative to

	the start of task
$\mu_{ke\theta}$	allocation/release coefficient of unit e in task k at time θ relative to the start of task
$\nu_{kr\theta}$	production/consumption of resource r in task k at time θ relative to the start of task
$\nu_{km\theta}$	production/consumption of material m in task k at time θ relative to the start of task
ν_m	value of product m
$\alpha_{ep'p''p\theta}$	allocation/release changeover coefficient of unit e from product p' to product p'' being at product p and at time θ relative to the start of the changeover task
ν_{km}	production/consumption of material m in task k
α_m	non-delivery penalty factor for product m
$\Theta_r^{min}, \Theta_r^{max}$	minimum and maximum amounts for product r
c_m^{raw}	cost of materials for product m
c_m^{sto}	cost of storage of material m
chg^{AP}	changeover duration
c_e	changeover cost in unit e
$c_{ep'p''}$	changeover time between product p' and product p'' in unit e
it_d^{AP}	length of delivery period d
$L_{rk}^{min}, L_{rk}^{max}$	minimum and maximum lot size of product r at campaign task k
NT_k^{fixed}	number of times task k runs (used in the rolling horizon approach)
Q_{md}^{DSmin},Q_{md}^{DSmax}	minimum and maximum amount of product m for delivery d
R_{rt}^{max}	maximum resource availability of resource r at time interval t
R_{mt}^{max}	maximum material m availability at time interval t
R_{md}^{max}	maximum resource availability of material m at delivery d
R_m^{APinit}	material m availability in the beginning of the planning horizon
R_r^{init}	resource r availability in the beginning of the scheduling horizon
R_e^{init}	unit e availability in the beginning of the scheduling horizon

R_m^{init}	material m availability in the beginning of the scheduling horizon
T	cycle time
T^{DS}	length of the scheduling horizon
t_d	time interval of the delivery period *d*
$V_{ke}^{min}, V_{ke}^{max}$	minimum and maximum capacity of unit *e* for task *k*
W	raw materials
W_l	raw materials associated to campaign l

Variables

Π_{mt}^{DS}	continuous variables that define the delivery of product *m* at time interval *t*
$\Pi_{md}^{DSslack}$	continuous variables that define the slack of product *m* at delivery *d*
Π_{md}^{AP}	continuous variable that define the amount of product *m* delivered at period *d*
$\Pi_{md}^{APslack}$	continuous variable that define the slack of product *m* at delivery *d*
Δ_r	continuous variables that define the net production of resource *r*
$R_r^{init'}$	continuous variables that define the resource *r* availability in the beginning of the scheduling horizon
R_{rt}	continuous variables that define the resource availability *r* at time interval *t*
R_{pet}^{DS}	continuous variables that define the resource availability *r* of product *p* at time interval *t*
R_{pe}^{init}	allocation of unit *e* at the beginning of the scheduling horizon
R_{mt}^{DS}	continuous variables that define the material availability m at time interval t
R_{md}^{AP}	continuous variables that define the availability of material *m* at delivery *d*
R_m^{APinit}	continuous variables that define the material *m* availability in the beginning of the planning horizon (used in the rolling horizon)
ξ_{kt}, ξ_{kt}^{DS}	continuous variables that define the batch size of task *k* at time interval *t*

ξ_{kd}^{AP}	continuous variables that define the total amount of material processed by task *k* at delivery *d*
$C_{ep'p''t}^{DS}$	binary variables that define the changeover task in unit *e* between product p' and product p'' and at time interval t
N_{kt}, N_{kt}^{DS}	binary variables that define if task *k* starts at time interval *t*
N_{k}^{DS}	integer variables that define the number of times task *k* runs
NC_{p}^{DS}	integer variables that define the number of changeovers associated to product *p*
N_{kd}^{AP}	integer variables that define the number of occurrences of task *k* at delivery *d*
Y_{ped}^{AP}	binary variables that define if product *p* is produced in unit *e* at delivery period *d*
λ_r	binary variables that define the selection of the storable intermediaries

Appendix A – Problems Data

Table 6.6 – Demand in tons for the illustrative example.

Weeks	1		2		3		4		5		6	
Product	min	max	min	max	min	max	min	max	min	max	min	max
P1	5	20	20	30	5	40	0	0	5	15	0	0
P2	0	20	10	10	10	30	20	30	10	30	15	30
P3	0	30	0	0	10	30	10	20	5	10	5	20
Weeks	7		8		9		10		11		12	
Product	min	max	min	max	min	max	min	max	min	max	min	max
P1	0	0	20	30	0	20	15	25	20	40	10	20
P2	5	5	10	40	10	60	5	10	5	15	10	20
P3	0	0	15	40	0	0	20	30	0	30	10	20

Table 6.7 – Demand in kg for the case study.

Weeks	1		2		3		4		5		6	
Product	min	max	min	max	min	max	min	max	min	max	min	max
PA	0	0	200	250	0	0	200	300	200	300	200	300
PB	200	360	200	360	200	360	200	360	0	0	200	360
PC	0	0	0	0	70	140	0	0	0	0	0	0
PD	0	0	0	0	180	260	0	0	0	0	0	0
PE	0	0	0	0	0	0	140	200	0	0	0	0
Weeks	7		8		9		10		11		12	
Product	min	max	min	max	min	max	min	max	min	max	min	max
PA	0	0	200	300	0	0	200	300	0	0	200	300
PB	200	360	200	360	0	0	360	480	0	0	360	480
PC	0	0	200	220	0	0	100	120	0	0	0	0
PD	0	0	200	220	0	0	0	0	0	0	200	220
PE	0	0	160	180	0	0	140	160	0	0	140	160

Table 6.8 – Processing units' characteristics for the case study.

Unit	Max. Volume	Min. Volume
U1	4000	100
U2	6300	150
U3	10000	50
U4	1000	100
U5	1300	50
U6	1000	50
U7	7000	130
U8	4000	80
U9	6300	150
U10	4000	120
F1	800	50
F2	500	30
D1	900	100
D2	600	100
V1	1000	100

Table 6.9 – Products value and raw material costs for the case study (m.u. –monetary units).

	Economic value [m.u]	Raw material cost [m.u/kg]
PA	10	5
PB	20	3
PC	15	6
PD	30	11
PE	70	36

Appendix B – Solution Statistics

Table 6.10 – Illustrative example: four weeks schedule with campaign tasks.

Model/horizon/aggregation	Int. variables/cont. variables/constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
AP/4W/C	48/85/204	0	51	2,303.09	0.00	2,090.0	0.3
DS/4W/C	1785/2317/3600	22,606	1,409,338	2,243.73	0.01	1,962.3	21.3
DS1/4W/C	1788/2317/3606	10,508	693,224	2,243.73	0.01	1,962.3	16.9
DSp1/4W/C	1788/2317/3606	53,784	5,283,160	2,243.73	0.01	1,962.3	55.0
DS2/4W/C	1791/2317/3609	16,471	1,190,734	2,243.73	0.01	1,962.3	22.3
RH1/4W/C	1785/2317/3623	0	911	2,009.23	0.00	1,962.3	5.4
RH2/4W/C	1785/2317/3602	12,666	617,703	2,191.95	0.01	1,962.3	13.9

Table 6.11 – Illustrative example: eight weeks schedule with campaign tasks.

Model/horizon/aggregation	Int. variables/cont. variables/constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
AP/8W/C	96/169/408	0	120	4,766.13	0.00	4,397.5	0.3
DS/8W/C	3549/4597/7152	1,027,859	115,570,751	4,630.48	0.01	4,121.5	2,165.0
DS1/8W/C	3552/4597/7158	755,217	64,563,162	4,630.48	0.01	4,121.5	1,133.3
DSp1/8W/C	3552/4597/7158	694,648	160,640,734	4,630.48	2.01	4,121.5	3,601.3
DS2/8W/C	3555/4597/7161	639,363	56,872,981	4,630.48	0.01	4,121.5	956.2
RH1/8W/C	3549/4597/7223	3,953	114,599	3,991.50	0.01	3,918.2	9.7
RH2/8W/C	3549/4597/7202	15,207	749,948	4,030.27	0.00	3,949.2	36.4

Table 6.12 – Illustrative example: eight weeks schedule with campaign tasks and different rolling horizon windows.

Model/horizon/aggregation	Int. variables/cont. variables/constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
RH1.1/8W/C	3549/4597/7223	2,370	82,848	4,146.74	0.00	4,046.4	21.9
RH2.1/8W/C	3549/4597/7203	10,083	650,502	4,226.94	0.00	4,081.3	47.3
RH1.2/8W/C	3549/4597/7198	46,449	2,734,607	4,228.89	0.01	4,040.5	57.4
RH2.2/8W/C	3549/4597/7180	23,631	2,245,720	4,387.78	0.01	4,112.2	63.7

Table 6.13 – Illustrative example: twelve weeks schedule.

Model/horizon	Int. variables/cont. variables/constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
AP/12W	216/433/900	10,528	149,206	7,561.25	0.01	6,973.1	2.0
DS/12W	6831/9913/16782	219,941	70,867,983	6,971.51	6.27	6,150.1	3,602.0
DS1/12W	6840/9913/16800	185,686	67,646,537	6,971.51	7.10	6,156.4	3,602.2
DSp1/12W	6840/9913/16800	89,491	49,004,502	6,971.51	14.71	5,805.0	3,601.8
DS2/12W	6843/9913/16803	142,460	57,272,458	6,971.51	8.16	6,062.6	3,601.9
RH1/12W	6831/9913/17085	10,610	595,273	6,021.66	0.01	5,806.1	89.9
RH2/12W	6831/9913/17033	13,688	1,384,723	6,072.89	0.00	5,806.1	177.4
AP/12W/C	144/253/612	191	1,063	7,142.23	0.01	6,538.7	0.8
DS/12W/C	5313/6877/10704	507,264	87,286,615	6,969.79	4.85	6,182.8	3,395.0
DS1/12W/C	5316/6877/10710	603,294	89,337,815	6,969.79	5.64	6,100.2	3,602.1
DSp1/12W/C	5316/6877/10710	270,009	82,649,307	6,969.79	7.77	6,070.3	3,601.8
DS2/12W/C	5319/6877/10713	402,645	69,102,174	6,969.79	6.78	6,135.3	3,601.8
RH1/12W/C	5313/6877/10817	931	20,258	6,045.70	0.01	5,888.8	14.3
RH2/12W/C	5313/6877/10800	8,708	368,738	6,007.75	0.01	5,808.4	28.9

Table 6.14 – Illustrative example: twelve weeks schedule with campaign tasks and different rolling horizon windows.

Model/horizon /aggregation	Int. variables/cont. variables/constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
RH1.1/12W/C	5313/6877/10809	10,834	636,166	6,298.37	0.00	6,053.7	28.4
RH2.1/12W/C	5313/6877/10793	18,487	1,448,855	6,348.70	0.01	6,059.2	83.4
RH1.2/12W/C	5313/6877/10796	7,907	640,003	6,397.02	0.00	6,124.9	45.2
RH2.2/12W/C	5313/6877/10778	45,417	3,067,954	6,441.88	0.01	6,162.1	132.9

Table 6.15 – Illustrative example: twelve weeks schedule without changeovers.

Model/horizon/aggregation	Int. variables/cont. variables/ constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
AP/12W	108/433/684	0	50	7,651.25	0.00	7,394.8	0.2
DS/12W	2277/8386/15261	0	3,704	7,161.65	0.00	7,161.7	0.5

Table 6.16 – Case study: eight weeks schedule.

Model/horizon/aggregation	Int. variables/cont. variables/constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
AP/8W	888/1673/3736	191,528	9,010,675	75,124.20	0.01	73,162.4	234.5
DS/8W	29913/37605/66577	40,970	41,645,010	74,239.90	39.33	52,082.6	14,439.9
DS1/8W	29978/37605/66707	30,153	52,636,580	74,239.90	32.30	54,873.5	14,410.6
DSp1/8W	29978/37605/66707	30,214	52,706,062	74,239.90	32.30	54,873.5	14,410.7
DS2/8W	29983/37605/66712	38,707	57,944,461	74,239.90	45.75	49,799.2	14,409.8
RH1/8W	29913/37605/66821	17,306	18,499,174	64,136.60	5.99	58,683.1	4,174.9
RH2/8W	29913/37605/66875	10,514	17,955,601	67,495.30	6.06	62,723.5	7,043.4
RH1.1/8W	29913/37605/66813	6,190	4,710,039	64,804.60	4.99	60,731.5	920.7
RH2.1/8W	29913/37605/66815	10,820	13,144,695	56,595.80	6.07	52,366.7	7,330.9
AP/8W/C	776/1225/3000	73,152	2,939,395	73,629.80	0.01	72,245.6	34.3
DS/8W/C	27547/31183/53719	31,175	51,628,895	70,684.50	34.92	51,748.6	14,434.8
DS1/8W/C	27598/31183/53821	27,502	58,879,606	70,684.50	42.33	49,140.8	14,405.4
DSp1/8W/C	27598/31183/53821	27,274	58,684,118	70,684.50	48.10	47,228.0	14,406.0
DS2/8W/C	27603/31183/53826	22,846	48,608,967	70,684.50	31.24	53,212.2	14,444.7
RH1/8W/C	27547/31183/53861	11,324	9,035,461	61,884.80	5.00	54,195.3	1,648.2
RH1.1/8W/C	27547/31183/53878	24,206	19,340,491	63,654.80	4.80	58,025.2	5,858.9
RH2/8W/C	27547/31183/53877	20,922	15,976,374	65,922.00	4.98	60,048.9	4,020.3
RH2.1/8W/C	27547/31183/53883	27,601	26,477,756	64,161.40	5.99	58,641.7	7,493.2

Table 6.17 – Case study: twelve weeks scheduling.

Model/horizon/aggregation	Int. variables/cont. variables/constraints	Nodes	Iterations	LP relaxation	Gap (%)	Objective	CPU time (sec)
AP/12W	1332/2509/5604	492,694	35,230,261	110,455.00	0.01	107,688.0	471.7
RH1/12W	44781/56273/100159	625	141,187	89,929.90	1.73	87,907.5	1,473.9
RH1.1/12W	44781/56273/100106	7,125	3,039,419	89,763.10	3.97	85,504.6	2,587.2
RH2/12W	44781/56273/100181	1,286	1,069,626	95,020.30	2.94	91,140.1	4,653.4
RH2.1/12W	44781/56273/100061	8,370	6,191,209	96,480.00	3.86	91,909.2	5,491.0
AP/12W/C	1164/1837/4500	70,965.00	3,798,482	108,357.00	0.01	106,446.0	56.63
RH1/12W/C	41239/46659/80744	190	97,528	85,952.10	4.90	81,790.3	2,788.05
RH1.1/12W/C	41239/46659/80667	3,933	1,112,363	87,456.00	4.86	80,867.7	7,555.55
RH2/12W/C	41239/46659/80713	2,887.00	1,287,263	86,198.00	1.70	82,639.4	4,419.79
RH2.1/12W/C	41239/46659/80645	5,592	2,878,518	86,647.10	2.81	83,801.4	8,203.44

References

Amaro, A. C. & Barbosa-Póvoa A.P.F.D. (2008), Optimal Supply Chain Management with Detailed Scheduling, Ind. Eng. Chem. Res., 47 (1), 116 -132.

Amaro A. C. & Barbosa-Póvoa A.P.F.D. (2008)b, Planning and Scheduling of Industrial Supply Chains with Reverse Flows: A real pharmaceutical case-study, Computers & Chemical Engineering, 32, 2606-2635

Barbosa-Povoa, A. P. (2007). A critical review on the design and retrofit of batch plants. Computers & Chemical Engineering, 31, 833-855.

Bassett, M., Dave, P., Doyle, F., Kudva, G., Pekny, J., Reklaitis, G., Subrahmanyam, S., Miller, D., & Zentner, M. (1996). Perspectives on model based integration of process operations. Computers & Chemical Engineering, 20, 821-844.

Bassett, M. H., Pekny, J. F., & Reklaitis, G. V. (1996). Decomposition techniques for the solution of large-scale scheduling problems. AIChE Journal, 42, 3373-3387.

Castro, P., Barbosa-Povoa, A., & Matos, H. (2001). An improved RTN continuous-time formulation for the short-term scheduling of multipurpose batch plants. Industrial & engineering chemistry research, 40, 2059-2068.

Castro, P., Barbosa-Povoa, A. P., & Matos, H. A. (2003). Optimal periodic scheduling of batch plants using RTN-based discrete and continuous-time formulations: A case study approach. Industrial & engineering chemistry research, 42, 3346-3360.

Castro, P., Novais, Q., & Carvalho, A. (2008). Optimal equipment allocation for high plant flexibility: An industrial case study. Industrial & engineering chemistry research, 47, 2742-2761.

Dimitriadis, A., Shah, N., & Pantelides, C. (1997). RTN-based rolling horizon algorithms for medium term scheduling of multipurpose plants. Computers & Chemical Engineering, 21, S1061-S1066.

Erdirik-Dogan, M., & Grossmann, I. E. (2007). Planning models for parallel batch reactors with sequence-dependent changeovers. AIChE Journal, 53, 2284-2300.

Fumero, Y., Corsano, G., & Montagna, J. M. (2012). Scheduling of Multistage Multiproduct Batch Plants Operating in a Campaign-Mode. Industrial & engineering chemistry research, 51, 3988-4001.

Ierapetritou, M., & Floudas, C. (1998). Effective continuous-time formulation for short-term scheduling. 1. Multipurpose batch processes. Industrial & engineering chemistry research, 37, 4341-4359.

Janak, S. L., Lin, X., & Floudas, C. A. (2004). Enhanced continuous-time unit-specific event-based formulation for short-term scheduling of multipurpose batch processes: Resource constraints and mixed storage policies. Industrial & engineering chemistry research, 43, 2516-2533.

Kondili, E., Pantelides, C., & Sargent, R. (1993). A general algorithm for short-term scheduling of batch operations--I. MILP formulation. Computers & Chemical Engineering, 17, 211-227.

Li, X., Gao, L., Zhang, C., & Shao, X. (2010). A review on integrated process planning and scheduling. International Journal of Manufacturing Research, 5, 161-180.

Li, Z., & Ierapetritou, M. (2008). Process scheduling under uncertainty: Review and challenges. Computers & Chemical Engineering, 32, 715-727.

Lin, X., Floudas, C. A., Modi, S., & Juhasz, N. M. (2002). Continuous-time optimization approach for medium-range production scheduling of a multiproduct batch plant. Industrial & engineering chemistry research, 41, 3884-3906.

Maravelias, C. T., & Grossmann, I. E. (2003). New general continuous-time state-task network formulation for short-term scheduling of multipurpose batch plants. Industrial & engineering chemistry research, 42, 3056-3074.

Maravelias, C. T., & Sung, C. (2009). Integration of production planning and scheduling: Overview, challenges and opportunities. Computers & Chemical Engineering, 33, 1919-1930.

Mauderli, A., & Rippin, D. (1979). Production planning and scheduling for multi-purpose batch chemical plants. Computers & Chemical Engineering, 3, 199-206.

Mendez, C. A., Cerda, J., Grossmann, I. E., Harjunkoski, I., & Fahl, M. (2006). State-of-the-art review of optimization methods for short-term scheduling of batch processes. Computers & Chemical Engineering, 30, 913-946.

Méndez, C., Henning, G., & Cerda, J. (2001). An MILP continuous-time approach to short-term scheduling of resource-constrained multistage flowshop batch facilities. Computers & Chemical Engineering, 25, 701-711.

Moniz, S., Barbosa-Póvoa, A. P., & Pinho de Sousa, J. (2012). Regular and non-regular production scheduling of multipurpose batch plants. Proceedings of the 22nd European Symposium on Computer Aided Process Engineering.

Pantelides, C. C. (1994). Unified frameworks for optimal process planning and scheduling. In (pp. 253-274): Cache Publications New York.

Papageorgiou, L. G., & Pantelides, C. C. (1993). A hierarchical approach for campaign planning of multipurpose batch plants. Computers & Chemical Engineering, 17, S27-S32.

Papageorgiou, L. G., & Pantelides, C. C. (1996a). Optimal campaign planning/scheduling of multipurpose batch/semicontinuous plants. 1. Mathematical formulation. Industrial & engineering chemistry research, 35, 488-509.

Papageorgiou, L. G., & Pantelides, C. C. (1996b). Optimal campaign planning/scheduling of multipurpose batch/semicontinuous plants. 2. A mathematical decomposition approach. Industrial & engineering chemistry research, 35, 510-529.

Pinto, T., Barbosa-Póvoa, A. P. F. D., & Novais, A. Q. (2005). Optimal design and retrofit of batch plants with a periodic mode of operation. Computers & Chemical Engineering, 29, 1293-1303.

Pochet, Y., & Warichet, F. (2008). A tighter continuous time formulation for the cyclic scheduling of a mixed plant. Computers & Chemical Engineering, 32, 2723-2744.

Reklaitis, G. (1995). Scheduling approaches for the batch process industries. ISA Transactions, 34, 349-358.

Schilling, G., & Pantelides, C. (1996). A simple continuous-time process scheduling formulation and a novel solution algorithm. Computers & Chemical Engineering, 20, S1221-S1226.

Schilling, G., & Pantelides, C. (1999). Optimal periodic scheduling of multipurpose plants. Computers & Chemical Engineering, 23, 635-655.

Shah, N., Pantelides, C., & Sargent, R. (1993). Optimal periodic scheduling of multipurpose batch plants. Annals of Operations Research, 42, 193-228.

Shah, N. (2004). Pharmaceutical supply chains: key issues and strategies for optimisation. Computers & Chemical Engineering, 28, 929-941.

Stefansson, H., Sigmarsdottir, S., Jensson, P., & Shah, N. (2011). Discrete and continuous time representations and mathematical models for large production scheduling problems: A case study from the pharmaceutical industry. European Journal of Operational Research.

Sundaramoorthy, A., & Maravelias, C. T. (2011). Computational Study of Network-Based Mixed-Integer Programming Approaches for Chemical Production Scheduling. Industrial & engineering chemistry research.

Sundaramoorthy, A., & Karimi, I. (2004). Planning in pharmaceutical supply chains with outsourcing and new product introductions. Industrial & engineering chemistry research, 43, 8293-8306.

Velez, S., & Maravelias, C. T. (2013). Reformulations and branching methods for mixed-integer programming chemical production scheduling models. Industrial & engineering chemistry research, 52, 3832-3841.

Verderame, P. M., Elia, J. A., Li, J., & Floudas, C. A. (2010). Planning and scheduling under uncertainty: a review across multiple sectors. Industrial & engineering chemistry research, 49, 3993-4017.

Vooradi, R., & Shaik, M. A. (2012). Improved Three-Index Unit-Specific Event-Based Model for Short-Term Scheduling of Batch Plants. Computers & Chemical Engineering.

Wu, D., & Ierapetritou, M. (2004). Cyclic short-term scheduling of multiproduct batch plants using continuous-time representation. Computers & Chemical Engineering, 28, 2271-2286.

Wu, D., & Ierapetritou, M. (2007). Hierarchical approach for production planning and scheduling under uncertainty. Chemical Engineering and Processing: Process Intensification, 46, 1129-1140.

Wu, D., & Ierapetritou, M. G. (2003). Decomposition approaches for the efficient solution of short-term scheduling problems. Computers & Chemical Engineering, 27, 1261-1276.

You, F., Castro, P. M., & Grossmann, I. E. (2009). Dinkelbach's algorithm as an efficient method to solve a class of MINLP models for large-scale cyclic scheduling problems. Computers & Chemical Engineering, 33, 1879-1889.

Zentner, M., Pekny, J., Reklaitis, G., & Gupta, J. (1994). Practical considerations in using model-based optimization for the scheduling and planning of batch/semicontinuous processes. Journal of Process Control, 4, 259-280.

7 Conclusions and Future Research

This thesis presents some new models and resolution approaches for the scheduling problem in multipurpose batch plants, as part of the development of a broader scheduling methodology. The primary objective of this work was to develop a general and integrated methodology for these complex, highly combinatorial problems. A real case-study from the chemical pharmaceutical industry was used as test-bed in this research. Emphasis was given to the specific features of this industrial sector, involving a significant work to contextualize and determine how the planning and scheduling functions are performed. The key contributions of this thesis and recommendations for future research are summarized in what follows.

7.1 Main Contributions of the Thesis

Scheduling problems in process industries have received considerable attention in the past decades due to their importance for the efficiency of operations. A variety of modeling approaches has appeared in the literature, introducing different types of formulations and involving multiple decisions and objectives. In general, there has been an effort to take into account the computational efficiency of the formulations, particularly when addressing large-scale scheduling problems. Nevertheless, modeling, computational performance, and the integration of optimization methods in the real decision-making processes of companies, are still open issues that have been addressed in this study.

In summary, this thesis: a) presents the Delivery Tradeoffs Matrix to expose the tradeoffs occurring in the drug development cycle; b) introduces the equipment redesign problem, which in practice permits more flexibility to the task-unit assignment decisions; c) proposes a new formulation to efficiently deal with sequence-dependent changeovers and temporary storage in the processing units (two complicating requirements of the discrete-time formulations); d) provides a new methodology that integrates the problem representation, the scheduling model, and the decision making process; and e) proposes non-exact methods based on a task-unit aggregation and time-based decompositions for solving large scale instances. In more detail, the contributions of each chapter are described below.

Chapter 2 redefines the planning and scheduling functions for the context of the chemical-pharmaceutical industry. Addressing design and scheduling decisions simultaneously is particularly interesting for executing processes that are under development. Moreover, we suggest a conceptual representation of the tradeoffs occurring in the drug development cycle, named the Delivery Tradeoffs Matrix.

Chapter 3 introduces the scheduling problem with equipment redesign. This problem has to do with performing changes in the processing units (mainly reactors) such that those units are capable of performing additional tasks. Allowing for changing the equipment in such way that the task-unit suitability is increased, can be viewed as an innovative approach to increase flexibility of batch plants.

Chapter 4 presents another contribution of practical importance. An efficient and general MILP discrete-time formulation for scheduling of multipurpose batch plants has been developed, that explicitly models the inventory carried out in each task. Following this modeling strategy, some aspects that might be quite complicated for discrete-time formulations, such as sequence-dependent changeovers and temporary storage in the processing units, can be modeled through new types of constraints, leading to very efficient models. Moreover, several other requirements that are common in the chemical and biochemical-pharmaceutical industries have been taken into account. These requirements, somehow neglected by existing formulations, include lots blending and lots traceability, alternative task-unit allocation, and task-unit-layout assignment.

Chapter 5 proposes a new methodology for addressing and solving scheduling problems in chemical batch plants. The developed conceptual framework can be seen as an innovative way to integrate the representation of the scheduling problem, the optimization model, and the decision-making process, in a coherent methodology to be used across several departments. The proof-of-concept of the methodology was performed in the case-study company, and demonstrated its applicability under realistic production scenarios. Moreover from a theoretical point of view, the proposed methodology can be a good starting point for significant, further developments.

Finally, Chapter 6 proposes differentiated aggregation levels for campaign and short-term scheduling. This solution approach contrasts with existing methods since, using the same model, it delivers schedules with periodic and non-periodic patterns. In practice, the approach can be used for the medium-term scheduling of batch plants in which production resources are shared between campaign and short-term operation modes, thus improving the system responsiveness. The developments were based on the RTN formulation, and to tackle large scale problems a rolling horizon approach, reformulation and branching strategies have been introduced.

The outputs of this thesis are significant contributions for better modeling scheduling problems and for solving real world problems in chemical batch plants. They can also be viewed as a sound basis for the development of improved and more sophisticated decision support tools for dealing with those problems.

7.2 Recommendations for Future Research

The ideas presented in this thesis point to several interesting research developments in the area. Five research opportunities are outlined in what follows, reflecting possible improvements on the current work or new research topics. The first research opportunity is related to the scheduling problem with equipment redesign. The second is concerned with the development of better aggregation formulations as part of the rolling horizon algorithm. The third is associated to processes scale-up strategies. The fourth proposes a hierarchical planning and scheduling approach. Finally, a fifth opportunity is related to the characterization of the uncertainty in the scheduling model.

The Scheduling Problem with Equipment Redesign

An efficient use of the processing units is fundamental to decrease the operational costs and increase the responsiveness of the manufacturing system. Thus, it would very interesting to extend the formulation developed for equipment redesign to account for multiple equipment modifications, and to derive operational schedules considering those modifications and associated setups. Note that this approach contrasts with work in the literature that has been mainly focused on the determination of production schedules, where task-unit assignment is done considering a set of processing units with fixed characteristics or, alternatively, the problem is addressed from the design perspective, in either a retrofit or a grassroot design perspective.

Scheduling Solutions of Large Instances

In practice, decision-making processes need to address complex decisions, leading often to mathematical programming models with a very large number of 0-1 decision variables. The computational performance of models is then of extreme importance, since decision-making processes ask for models that are capable to deliver good quality solutions in very short times. The rolling horizon algorithm presented in chapter 6 has been designed to tackle this computational complexity, but it can still be improved by developing more accurate and still time-efficient aggregate formulations for the scheduling problem. Moreover, the development and assessment of (meta) heuristic procedures should also be explored, since these procedures should hopefully allow the determination of good solutions in very short computational times that could serve as warm start to a second stage optimization algorithm.

The Scheduling Problem and Scale-up Strategies

In the chemical-pharmaceutical industry, probably more than in other sectors, the product is strongly linked to the process. The way chemical processes are designed, implemented, and scaled-up strongly determine the overall cost of the product, the total production time, and also the global efficiency of the multipurpose batch plant. In this context, an interesting question arises: *how to define optimal scale-up strategies taking into account the optimal utilization of the multipurpose batch plant?* Answering this question may require modeling scale-up decisions overtime and the development of efficient formulations / solution approaches that select the adequate processing units in the course of successive process scale-ups. Although the scheduling model proposed in this thesis (see Chapter 4) accounts for lot size decision variables, a holistic approach capable of addressing the long–term dimension of the scale-up decisions and the multi-objective nature of the problem is a step that should be pursued.

Hierarchical Planning and Scheduling

This thesis addresses the scheduling problem considering several time horizons (short and medium-term). An immediate consequence of this strategy is that decisions are taken in just one decision level. Another way to address the problem would be to identify the different types of decisions, in a hierarchic way. For example, in a multi-site manufacturing system, as is the case of the company addressed in this study, the decisions on the allocation of products to plants would be made first than the task-unit assignment decisions. Or even, tasks-unit assignment and task-unit sequencing could be performed in sequence. In this way, a hierarchical approach would frame planning and scheduling decisions, according to the company's decision-making principles, this surely leading to a significant decrease of the computational burden.

Dealing with Uncertainty

The scheduling problem addressed in this thesis has been considered as deterministic. Although this seems to be as reasonable assumption, allowing for the determination of realistic scheduling solutions, uncertainty may be present in some important parameters. Uncertainty is mainly associated to the processing times of tasks of the products under development and to the demand. The extension of the formulation present in Chapter 4 to deal with uncertainty will surely be of great practical relevance.