1 2 9 0

UNIVERSIDADE Ð COIMBRA

José Francisco Vaz Suspiro

REPRESENTATION OF CHEMICAL PROCESSES THROUGH DISCRETE-EVENT SIMULATION

Dissertação no âmbito do Mestrado Integrado em Engenharia Mecânica orientada pelo Professor Doutor Samuel Moniz e apresentada no Departamento de Engenharia Mecânica da Universidade de Coimbra

Fevereiro de 2020

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE D COIMBRA

Representation of chemical processes through discrete-event simulation

Submitted in Partial Fulfilment of the Requirements for the Degree of Master in Mechanical Engineering in the speciality of Production and Project

Representação de processos químicos através de simulação de eventos discretos

Author
José Francisco Vaz Suspiro
Advisor
Samuel de Oliveira Moniz

Jury

President	**Professor Doutor Luís Miguel D. F. Ferreira** **Professor da Universidade de Coimbra**
Vowels	**Professor Doutor Telmo Miguel Pires Pinto** **Professor da Universidade de Coimbra**
Advisor	**Professor Doutor Samuel de Oliveira Moniz** **Professor da Universidade de Coimbra**

Coimbra, February, 2020

"The enterprise that does not innovate ages and declines. And in a period of rapid change such as the present – the decline will be fast."

Peter Drucker, 2014.

To those near and dear

ACKNOWLEDGEMENTS

This dissertation was only made possible thanks the help and guidance of some people to whom I am deeply obliged.

My parents, who made me the person I am today and who supported and encouraged me to pursue whatever path I desired to take.

Professor Samuel Moniz, through whom I learned a lot, who kept me motivated and whose input was critical to the success of this work.

And to all my great friends, who were quite simply, awesome.

To you, my deepest gratitude.

Abstract

This work studies the dynamics of chemical processes from a well-known problem introduced by Kondili, Pantelides, & Sargent (1993). It aims to minimize the computational burden of optimization algorithms, by developing a discrete-event simulation model that portraits some system characteristics that the analytic methods have difficulties capturing.

The problem is modeled considering several structural aspects of the plants, as well as detailed process characteristics such as material transfer rates and material transfer logistics. The results of the optimization model that do not consider these system characteristics are then contrasted with the estimates obtained by the simulation model.

Furthermore, the model was formulated in a recipe-oriented way that allows for the simulation of a wide variety of different batch chemical processes with minor model adjustment.

The simulation model proved to be computationally efficient and, when compared with the optimization model, delivered better total production duration and intermediate storage profiles estimates.

Keywords Discrete-event Simulation, Mixed Integer Linear Programming, Process design, Short-term scheduling.

Resumo

Este trabalho estuda a dinâmica de processos químicos de um problema conhecido introduzido por Kondili et al. (1993). Com vista a minimizar o esforço computacional do modelo de otimização, um modelo de simulação de eventos discretos foi desenvolvido, retratando algumas características do sistema que o modelo de programação matemática tem dificuldade de capturar.

O problema é modelado considerando vários aspetos estruturais das instalações industriais, bem como características detalhadas do processo, como taxas de transferência de material e a logística de transferência de material. Os resultados do modelo de otimização que não consideram essas características do sistema são contrastados com as estimativas obtidas pelo modelo de simulação.

Além disso, o modelo foi formulado com base no plano de produção de forma a permitir a simulação de uma variedade ampla de instalações químicas com produção em lote, sem que haja a necessidade de ajustes significativos.

O modelo de simulação mostrou-se eficiente em termos computacionais e, quando comparado com o modelo de otimização, proporcionou melhores estimativas da duração total de produção e dos perfis de utilização de armazenamento intermediário.

Palavras-chave: Simulação de eventos discretos, Programação Linear Inteira Mista, Projeto de processos, Planeamento de curto prazo.

Contents

LIST OF FIGURES ix
LIST OF TABLES xi
LIST OF SYMBOLS AND ACRONYMS/ ABBREVIATIONS xiii
List of Symbols xiii
Acronyms/Abbreviations xiii
1. Introduction 1
1.1. Motivation 1
1.2. Concepts 2
1.2.1. Chemical Plants 2
1.2.2. Mathematical optimization 3
1.2.3. Discrete-event simulation 4
1.3. Objective 4
1.4. Methodology 5
1.5. Structure of the dissertation 6
2. Literature review 7
2.1. Optimization 7
2.2. Simulation 10
2.3. Optimization-Simulation approaches 12
3. An optimization-simulation approach 15
3.1. Problem description 16
3.2. Case Study 16
3.2.1. Chemical process 17
3.2.2. Plant equipment and storage 17
3.3. Optimization model 18
3.3.1. Mathematical formulation 19
3.3.2. Optimization results 21
3.3.3. Analysis of optimization results 23
3.4. Simulation model 23
3.4.1. Selection of simulation software 24
3.4.2. Construction of the plant layout 25
3.4.3. Development of Production plan 29
3.4.4. Programming the plant behaviour 31
3.4.5. Validation of the model 36
4. Approach assessment 39
4.1. Computational performance 39
4.2. Tank sizing 42
4.3. Production duration estimates 45
5. Conclusions 49
5.1. Further research 50

BIBLIOGRAPHY 51

LIST OF FIGURES

Figure 3.1. Process diagram (Kondili et al., 1993). 17
Figure 3.2. Production plan Gantt chart based on (Kondili et al., 1993). 22
Figure 3.3. Storage utilization profiles for IntBC, IntAB and ImpureE (Kondili et al., 1993). 22
Figure 3.4. A *FlowSource* creating material flow which is then deleted by a *FlowSink*. 26
Figure 3.5. The graphical representation of a *Tank* and a *Container Entity* 26
Figure 3.6. The graphical representation of *Pipes*, *Connectors* and *FlowNodes*. 27
Figure 3.7. The graphical representation of *ItemToFlowConverter* and *FlowToItemConverter*. 28
Figure 3.8. Case study plant representation in the graphical interface of SIMIO. 28
Figure 3.9. Example of a SIMIO process. 32
Figure 3.10. PPTs sub-types. 33
Figure 3.11. Representation of *ProcessExecuteTask* 35
Figure 3.12. IntBC storage utilization profile from literature (Kondili et al., 1993). 36
Figure 3.13. IntBC storage utilization profile according to the simulation model. 36
Figure 3.14. IntAB storage utilization profile from literature (Kondili et al., 1993). 37
Figure 3.15. IntAB storage utilization profile according to the simulation model. 37
Figure 3.16. ImpureE storage utilization profile from literature (Kondili et al., 1993). 37
Figure 3.17. ImpureE storage utilization profile according to the simulation model. 37
Figure 3.18. Product1 and Product2 storage utilization profile according to the simulation model. 38
Figure 4.1. Computational processing times of different production lengths. 42
Figure 4.2. IntBC Storage utilization profiles with different material transfer rates. 44
Figure 4.3. IntAB Storage utilization profiles with different material transfer rates. 45
Figure 4.4. Changes in logistical delay along different transfer rates. 47

LIST OF TABLES

Table 3.1. Task numeration. 17
Table 3.2. Plant equipment (Kondili et al., 1993). 18
Table 3.3. The different production plan columns and their respective information. 31
Table 4.1. The processing duration, half-width, and margin of error throughout different production intervals after 10 replications. 40
Table 4.2. The processing duration, half-width, margin of error and PR throughout different production intervals after 82 replications. 41
Table 4.3. Maximum tank weight [kgs] according to different transfer rates. 43
Table 4.4. Process duration at different transfer rates and throughout different production intervals. 46

LIST OF SYMBOLS AND ACRONYMS/ ABBREVIATIONS

List of Symbols

n – Necessary number of replications to achieve the desired margin of error

n_0 – Number of initial replications

h_0 – Half-width calculated from the initial set of replications

h – Target half width

Acronyms/Abbreviations

CPU – Computer processing unit

MILP – Mixed Integer Linear Programming

mSTN – Maximal State-Task Network

PPT – Production plan task

PR – Processing ratio

RTN – Resource-Task Network

STN – State-Task Network

1. INTRODUCTION

The short-term scheduling of chemical batch plants is critical to ensure superior levels of operational performance and overall competitiveness and profitability of the industry.

Thankfully, the same highly competitive global economy that forces companies to strive for the slightest operation improvement has also pushed computational technologies to evolve at a tremendous rate. This evolution allowed not only for the solvency of more significant and more convoluted problems using pre-existing technologies, but also for the emersion of tools such as discrete-event simulation as much more viable alternatives in the optimization strategy spectrum.

This dissertation will focus on the use of discrete-event simulation as a parallel tool to optimization, used to minimize the shortcomings of these methods. This chapter will cover: i) the motivation for this study; ii) the concepts inherent to the problem; iii) the objectives of this dissertation; iv) the proposed methodology; and v) the structure of this dissertation.

1.1. Motivation

Production planning and scheduling problems are inherent to any flexible manufacturing process, and finding optimized solutions is critical for the viability and competitiveness of the organizations. In the global economy, companies operate in a highly competitive space, where even small increases in productivity are persistently pursued. Thanks to the constant advance of technology, new and better tools have emerged to tackle these problems and achieve a level of optimization previously unobtainable.

For the past few decades, optimization methods have been handy tools for effective production planning and efficient scheduling (Griva, Nash, & Sofer, 2009). These methods can obtain the best solutions according to the provided data and a set of constraints. On the other hand, though, it has also become quite clear that solving time for highly complex problems, especially using the more general algorithms, is unfeasible.

Very complex problems have been optimized in acceptable times (Samuel Moniz, Ana Paula Barbosa-Póvoa, 2013), but algorithm efficiency is usually inversely related with the algorithm's applicability, meaning the more efficient one gets, the narrower is the range of problems it can usefully solve. A possible approach to deal with this tradeoff is to "simplify" the problem. By modeling certain aspects or characteristics of the problem in an aggregated way, it is possible to reduce the complexity considerably, and therefore the resolution time.

The disadvantage of this approach, however, is that oversimplification of scheduling problems will lead to solutions that are not suitable for immediate utilization, and that must be calibrated. This calibration can be done by experienced workers, sometimes with help from actually production line tests. Though, due to exponential growth in computation power and problem complexity, discrete-event simulation has become a very viable and perhaps the best option for this kind of task.

In the last couple of decades, simulation has been used to great length, in a wide range of applications and proven very useful. The Athens Olympic Games Organizing Committee used it to optimize the logistics of the event, saving an estimated $69.7 million (Beis, Loucopoulos, Pyrgiotis, & Zografos, 2006). One year later, PMC used discrete-event simulation to create a "Demand-Driven Workforce Scheduler" to improve the logistics of a car renting company saving an estimated $80 million in labor reduction annually (Zottolo, Williams, & Ulgen, 2007).

This dissertation will focus on the utilization of discrete-event simulation as a tool for evaluating optimization solutions of multipurpose batch chemical plants, where various resources can be utilized to produce several products.

1.2. Concepts

1.2.1. Chemical Plants

In the chemical field, there are two types of manufacturing operations: batch and continuous tasks. Running a process in a continuous manner means that all steps of the chemical process are co-occurring, and typically the feedstock and output are also continuous. Continuous processes are very efficient on a large scale and are generally used to produce commodity chemicals (high demand chemicals) like acetone, ethanol or nitric acid (Sinnott & Towler, 2019).

On the other hand, specialty chemicals (low demand chemicals) such as fragrances, cleaning materials or paints, which do not sell in high volumes, have variable demand or whose formulation changes frequently, do not warrant the construction of a dedicated plant (Sinnott & Towler, 2019). For that reason, manufacturers prefer to produce this kind of chemicals using batch operation. The batch operation produces chemicals in a sequential way, i.e., in discrete amounts. The chemicals produced in each batch change often, and the main advantage is the flexibility of production.

There are two main types of batch chemical plants and each one with an increasingly higher level of flexibility, multiproduct and multipurpose plants. In multiproduct batch plants, each product has a preassigned equipment sequence, and only one product is produced at a time, except when there is a product change-over.

The plant considered later in this dissertation is the most flexible of the two, a multipurpose batch plant. This type of plant can produce multiple different products simultaneously, and the same product can be produced using different equipment units (Rippin, 1993).

1.2.2. Mathematical optimization

Optimization, as a matter of definition, is the act of making the best of something. Mathematical optimization achieves this goal through mathematical formulation and it has been used for decades in a wide variety of applications and continues to be the primary choice as a decision-making tool for complex problems (Griva et al., 2009).

To use this methodology, the problem variables must be identified, constrained according to the system characteristics, and an objective function must be constructed using the same variables (Nocedal & Wright, 2006). This function should be created in par with the intention of the optimization. For example, One might want to run a business by maximizing profit, and someone else might prefer to minimize loss, maximize efficiency, or minimize risk (Griva et al., 2009).

Once the problem has been modeled, it can be optimized by an optimization algorithm. There is an extensive amount of different optimization algorithms, each with its own advantages and disadvantages. This choice is of the utmost importance since it will decide whether or not the problem will be solved in a timely matter.

1.2.3. Discrete-event simulation

Simulation is a general concept, but in the context of operational research, simulation is best described as "the imitation of the operation of a real-world process or system over time" (Banks, Carson, Nelson, & Nicol, 2013).

Simulation is usually separated into two categories, continuous or discrete. Although systems are rarely entirely characterized by either one, they are frequently dominated by one or the other. Generally, the dominant type of system determines which type of simulation package should be used to reproduce it (Law, 2013).

Discrete-event simulation is the modeling of a system as it evolves at discrete time intervals, more specifically, it "concerns the modeling of a system as it evolves over time by a representation in which the state variables change instantaneously at separate points in time." (Law, 2013).

In opposition, continuous simulation is the modeling of systems whose state variables change continuously in time. This dissertation will focus on discrete-event simulation only.

1.3. Objective

The general purpose of this dissertation is to evaluate the use of discrete-event simulation as a parallel tool to mathematical optimization in a way that alleviates the processing burden from this method, allowing the analysis of more complex problems in acceptable time frames. More specifically, the objective is to create a methodology that can be used to implement this symbiosis in the optimization of chemical processes and evaluate its effectiveness.

To do so, a discrete-event simulation model of a complex chemical plant will be created where equipment, material transfer rates, and material transfer logistics. This model will run a production plan obtained from an optimization algorithm that did not consider these system dynamics.

Finally, the results of the optimization model will be compared with the simulation estimates to evaluate the proposed methodology.

1.4. Methodology

Optimization, in a more general way, is the action of achieving optimal values or solutions; however, as previously mentioned, there are some limitations that must be considered. In short, optimization can find the best possible results but tends to struggle with big problems, which usually leads to modeling simplification and, consequently, less meaningful optimal solutions.

Simulation, on the other hand, is well-equipped to achieve or prove feasibility, its utility stems mainly from scenario evaluation and variability implementation (Banks et al., 2013). Curiously, when comparing both, there is a perceptible symmetry of strengths and weaknesses. This is perhaps what makes the combination of both techniques a promising prospect.

Different approaches to the combination of these two methods have been taken throughout the last few decades, as reviewed by (Figueira & Almada-Lobo, 2014). The proposed methodology is categorized as SCS (Solution Completion by Simulation) and was based on the one described in the previously mentioned article. This type of approach aims to complete or correct the results obtained from the analytical method through the use of simulation.

More specifically, the results from a deterministic analytic model taken from (Kondili et al., 1993) will be enhanced by a detailed discrete-event simulation model created by the team of industrial management.

The case study and the implementation of the methodology will be present in Chapter 3.

1.5. Structure of the dissertation

This introductory chapter will be followed by a literature review of optimization, simulation, and the combined use of both. After the literature review, Chapter 3 will cover the case study, details about the optimization model, the simulation model, and the implementation of the proposed methodology. Next, in chapter 4, a thorough assessment of the results is provided, followed by the final chapter, in which conclusions will be drawn, and future work opportunities are enunciated.

.

2. LITERATURE REVIEW

As noted previously in 1.2.1, flexibility is the main advantage of a multipurpose batch chemical plant. However, it also means that production planning and scheduling will be much more complicated in comparison to a single product plant running in continuous operation.

A large assortment of problems of this kind has been solved throughout the years, through a wide variety of different methods. In the next subchapters, the most notable approaches will be separated by three categories: optimization, simulation and optimization-simulation, and will be briefly explained.

2.1. Optimization

Mathematical optimization is the main tool used to solve the previously mentioned problems. It was used as early as 1979 when a linear programming methodology was proposed to solve long term scheduling problems in multiproduct batch plants (Mauderli & Rippin, 1979). This methodology was useful when significant changes are introduced in manufacturing plants, like new equipment or new processes. However, short- and medium-term scheduling and process planning were left to an experienced engineer in charge of adapting the whole process according to the short-term chemical demand. At this time, the computing power was quite limited, meaning that optimization methods were only useful as long as it was possible to overcome the factorial growth of the solution space present in such processes.

One decade later, the complexity of the problems solved naturally increased, and so did the diversity of the solutions. Research contributions came in the form of a general algorithm for short-term scheduling of batch operations, based on a MILP formulation, by the name of State-Task Network (STN) (Kondili et al., 1993). This model could be utilized in a wide array of different short-term scheduling problems and had the main feature of differentiating between equipment and materials. This framework was later improved on resulting in a new variant of STN, called the Maximal State-Task Network (mSTN). This methodology combines STN and the necessary information about the process with information about the equipment and connectivity (Crooks C., 1992).

Later in 1994, intending to develop an accurate general-purpose scheduling algorithm and software, Pantelides (1994) advocated for the need to establish general frameworks for the representation of these problems. He proposed the use of two different, unified frameworks. A novelty called the Resource-Task Network (RTN) and the State-Task Network (STN). The main difference between them is that RTN considers a uniform representation of all resources and materials, and STN, on the other hand, differentiates the two.

In the same year, Barbosa-Póvoa & Macchietto (1994) argue that the design and scheduling aspects should be considered simultaneously. The benefit of this approach is the traditional sequential approach could be executed concurrently. Expanding on the ideas of Crooks C. (1992), the authors improve the mSTN formulation to be capable of optimizing structural aspects of the production schedule, while accounting for both the cost of equipment, pipework and operating costs and revenues.

In 2008, Pinto, Barbósa-Póvoa, & Novais (2008) made a comparative analysis between the STN, mSTN and RTN formulations and representations. In 2013, Samuel Moniz, Ana Paula Barbosa-Póvoa (2013) propose a new MILP discrete-time formulation based on STN, which explicitly models the inventory carried at the end of each task, among other features. This model is compared to an RTN model with a few integrated scheduling features present in literature. To carry out this comparison, the two formulations are tested in 4 different benchmark problems, 3 of them from literature and one original problem. It is concluded that the novel model was more computationally effective but that with the increase in problem size/complexity, both models struggle in proving optimality.

Companies operating in different regions, countries, and especially worldwide have built their facilities in key geographical locations to facilitate demand fulfillment in the different markets in which they operate (Papageorgiou, 2009). In the last decade, operating them as independent has become impossible if a company pretends to maintain competitiveness, sustainability, and growth. Papageorgiou (2009) presented a review of the of methodologies aimed to enhance infrastructure planning decision-making in the process industry. Three years later, Shah & Ierapetritou (2012) approached the scheduling for multisite, multipurpose batch plants using the augmented Lagrangian method. The analytic method optimized shipping costs optimizing production, inventory from a given fixed demand prediction.

Energy-saving has become a serious concern in the last years and has been incorporated in a wide variety of industrial plants, batch chemical plants included. To overcome the suboptimal results related to treating heat management, as an afterthought, (Seid & Majozi, 2014) created a framework that incorporates heat integration simultaneously with plant scheduling. The proposed framework has proven to have superior performance when compared to existing approaches, both in optimal values and processing time.

In 2017, Lee & Maravelias (2017) presented two discrete-time mixed-integer linear programming formulations for multipurpose batch plants. These representations are based on the STN and RTN that take into account limited intermediate storage. Other features such as limited shared resources, modeling of time-varying data, and linear modeling of inventory and resource utilization profiles were also discussed. Performance tests revealed that this methodology was suitable for medium-scale problems.

In 2018, Ackermann, Fumero, & M. Montagna (2018) presented a framework for simultaneous production planning and scheduling of multisite, multiproduct batch plants with nonidentical parallel units. The consideration of both problems simultaneously allows for the consideration of the critical tradeoffs between them. On the other hand, the problem becomes much more complicated, still examples were solved in reasonable times.

2.2. Simulation

Simulation has proven itself as an extremely useful tool in the operational research field to help solving production and logistic problems in a wide array of different fields, from communication networks to traffic systems. Short-term batch chemical plants are just another case where simulation can be beneficial (Jung, Blau, Pekny, Reklaitis, & Eversdyk, 2004).

As early as 1960, researchers have pointed to simulation as a guidance tool for the design of batch chemical plants. In this same year, Youle (1960) conducted the simulation of a hypothetical multi-stage batch chemical plant, the problem only considered one product, and the plant consisted of seventeen batch reactors distributed among four sequential reaction stages. The simulation algorithm was made for this single purpose and was based on discrete-event simulation. The author concluded the methodology to be useful in improving the factory’s efficiency, output and bottleneck detection.

At this time, the method seemed promising, but the available computing power made it impractical. It would not be until quite a few years later that it could be used effectively. The biggest challenge of using simulation in multipurpose batch chemical plants appeared to be its hybrid nature. Hybrid systems have elements of both discrete-event and continuous-time subsystems. This means that in these systems, some variables change in discrete time intervals while others, such as liquid flow, change continuously through time.

Fahrland (1970) was one of the first to write about the use of hybrid systems in simulation. In 1974, Fruit, Reklaitis, & Woods (1974) developed a hybrid simulation package for the simulation of mixed multiproduct batch and continuous processes called DISCON. The program's utility is demonstrated in a simplified polymerization facility where four cases are tested. Case 1 is the original plant, and the other three cases are the same original plant with some equipment upgrades. The simulation package proved very useful in the evaluation of equipment changes and bottlenecks.

In the following decade, still with particular purposes in mind, new hybrid simulation packages continued to appear. BOSS (Joglekar & Reklaitis, 1984) and

UNIBATCH (Czulek, 1988) are two of the most noteworthy, created to simulate batch/semi-continuous plants.

In 1994, Barton & Pantelides (1994) present the first general-purpose hybrid process modeling environment. This was a significant step in the evolution of simulation, since it allowed the user to focus on the modeling of the problem instead of wasting his time on the coding of the program itself. This program overcame the narrow usefulness of most simulation packages that operated in such manner up to the date. General-purpose simulation packages already existed up to this point, but the authors argued that few processes operate entirely in a continuous matter, meaning that a dynamic package would more accurately describe the processes.

In 1995, a hybrid simulation software was created to simulate multipurpose batch plants. The system could track the complete state of the plant and was based on the concept of recipe-controlled operation. This meant the system was separated into two parts: the plant and its functionality; and the recipe Engell, Fritz, Schulz, & Wöllhaf (1995).

Research into the topic seemed to suggest that for the type of problem considered in this dissertation, simulation alone would not be a particularly effective strategy. Simulation's strengths appear to be in scenario evaluation, and this methodology is not suited to find optimal values. For these reasons, further reviews of this topic will be done in tandem with optimization in the next chapter.

2.3. Optimization-Simulation approaches

At this present moment, it is clear that optimization models have the disadvantage of being computationally inefficient. Large problems tend to be difficult to solve by optimization on time while being adequately represented. Compromises must be made and the problem must be simplified, which will then produce solutions that are not perfectly calibrated for the problem at hand. On the contrary, simulation is a great tool to use in conjunction with optimization as a hybrid approach (Marques, Moniz, de Sousa, & Barbosa-Póvoa, 2017).

In 1994, Fu (1994) studied the use of different techniques for the optimization of stochastic discrete-event systems through the use of simulation. Both continuous and discrete parameter cases were considered. Perturbation analysis, likelihood method, and frequency domain expectation were the techniques focused on, for the former, and ranking-and-selection and multi-comparison procedures the ones focused on, for the latter.

In 1998, a hybrid methodology was proposed to solve the short-term batch scheduling of chemical or semiconductor plants. This methodology consists of two steps, the development of a discrete-event simulation model that can represent the production system behavior and as many of its inherent features, and the development of a general algorithm responsible for the batch process scheduling. After both the simulation model and the optimization algorithms are created, they will be used in an iterative process where the batch schedule is obtained by the analytic method and then introduced into the simulation model where an evaluation is made. After this, the objective function is adjusted accordingly and the cycle restarts. The cycle will be repeated until the scheduling solution is close to optimal. This approach yielded interesting solutions and reduced the search space considerably, especially with the increase in problem size. It was also noted that the bulk of CPU processing time was spent on the discrete-event simulation model (Azzaro-Pantel, Bernal-Haro, Baudet, Domenech, & Pibouleau, 1998).

In 1999, Byrne & Bakir (1999) solved a simple manufacturing system by combining analytic methods with simulation modeling in an iterative loop. The analytic model had difficulties in accommodating the system characteristics such as queuing and transportation delays, making the analytic solution dissatisfying. On the other hand, the simulation model could consider these characteristics, and when used in an iterative loop alongside the optimization algorithm, this hybrid procedure nullified the shortcomings of the analytic method allowing for a quasi-optimal solution to be found.

By the beginning of the new millennium, the conjunction of optimization and simulation had become widespread; at this point, almost all simulation packages had had some form of optimization. In 2000, the term “simulation optimization” was introduced in the *Encyclopedia of Operations Research & Management Science.*

In 2002, Fu (2002) made a comprehensive analysis of the optimization methods offered in discrete-event simulation packages and realized that there had been important gaps in the optimization-simulation research. Even though the offered optimization packages were a good start, they left a lot to be desired. The author argues the need for more general algorithms that can be used in a broader range of problems, more intuitive user interfaces that allow a wider user base, and improvements in efficiency to minimize computing time for large problems.

In multiproduct chemical batch plants there are many uncertainties to consider such as material prices, process failures, and others, but perhaps the most impactful is demand uncertainty, which can result in under or over-production. Intending to optimize inventory Jung, Blau, Pekny, Reklaitis, & Eversdyk (2004) used a hybrid simulation-optimization approach to solve this problem. To do so, stochastic discrete-event simulation of the supply chain is performed multiple times utilizing scheduling plans obtained from deterministic optimization methods for the complete problem horizon. The results of these simulations are utilized to estimate customer satisfaction and to adjust optimization parameters. This methodology is repeated in an iterative manner until the satisfaction levels reach the desired value.

With the aim of minimizing the expenses inherent of bringing a new drug to the market, Chen, Mockus, Orcun, & Reklaitis, (2012) developed a simulation-optimization methodology to improve management of the clinical supply chain in the pharmaceutical industry. The developed framework encompassed stochastic demand forecasting, a decentralized planning model and discrete-event simulation. This approach was demonstrated trough 3 different demand variations, from low to high demand. It proved very flexible when evaluating the tradeoff between customer service level and supply chain costs and an effective tool for risk management.

In 2016, Amaran, Sahinidis, Sharda, & Bury, (2016) made an extensive review of the algorithms and applications used to optimize stochastic simulation. Besides observing the topic was very rich in literature and of high interest in the area, the author found multiple topics that required investigation. Large-scale problems with combined discrete/continuous variables was one of them.

In the same year, Marques, Moniz, de Sousa, & Barbosa-Póvoa (2016) presented an innovative approach for product launch planning in the pharmaceutical industry, while considering the uncertainty of both the clinical trials and the product demand. A MILP analytical model with simulation incorporation was developed for the optimization of process design and capacity planning (such as process-unit allocation, scale-up decisions, and acquisition of new units) while taking into consideration future and actual commercialized products. This work was continued in (Marques et al., 2017) where it was further investigated and enhanced with the incorporation of features such as lot traceability, scale-up and clinical trials waste management. The proposed decision-making framework was tested on a case study inspired by a real situation. In a highly uncertain setting, the decision-making tool was able to effectively and clearly identify process configurations and scale-ups that maximize profit. Moreover, the results undeniably show the impact of uncertainty in the net present value reinforcing the inadequacy of deterministic models for real-life decision-making.

With the objective of managing the industrial planning and scheduling problem in an automated assembly line, Vieira, Barbosa-Póvoa, Moniz, & Pinto-Varela, (2018) successfully developed an iterative optimization simulation methodology solution capable of providing optimal production scheduling and number of robotics required. This author used a ROSA methodology, the same later used in this work.

3. AN OPTIMIZATION-SIMULATION APPROACH

Optimization has been extremely useful in solving a wide variety of complex problems. Mathematical optimization has the advantage of working in the full solution domain, meaning that it is possible to reach optimal results. Its main drawback, though, is the computational efficiency.

The higher the level of detail introduced in the algorithm, especially ones that originate complex constraints, the longer the model will take to find optimal results. Model efficiency can be improved, and complex problems can be modeled fully and still be capable of returning optimal solutions on acceptable time frames, the downside however, is that there is a clear correlation between model efficiency and its domain of usefulness. The more efficient an algorithm becomes, the narrower is the area in which it is suitable. Very efficient models tend only to be used in very particular situations and can be quite time consuming and expensive since it usually requires highly skilled personnel to create them.

It is generally more practical to use a general-purpose algorithm to optimize a particular problem. However, these algorithms tend to lead to oversimplified problem models, and the higher the level of simplification, the farther the results will be from reliable optimal results.

Discrete-event simulation has proven very useful to bridge the gap between the results obtained from a simplified algorithm and the accurate optimal results of a given problem. Capable of including problem features not considered in the analytical model and easily accounting for variability. Usually, this is done by the analysis of multiple simulation runs based on the optimization results that are then adjusted accordingly in an iterative process.

This chapter will go through all the steps of the presented investigative approach. It will start with a brief problem explanation, and then the case study is presented, followed by the description of the optimization methodology and the presentation of its results. Finally, this chapter ends with the description of the simulation model, its development and features.

3.1. Problem description

If a plant has access to long-term and accurate demand predictions, then it can run on “campaign mode”. In these situations, the plant resources are allocated over a long period of time. Planning and scheduling decisions are more straightforward since the need for changes in production is lower.

Alternatively, if product demand is not available, then production is highly dependent on available orders, forcing short-term scheduling. This is when a batch plant’s flexibility truly comes into play. With a small production horizon, fast scheduling and flexible resource allocation are the backbone of a profitable production.

In the next subchapters, a short-term scheduling problem of a multipurpose chemical batch plant is introduced.

3.2. Case Study

The problem approached in this dissertation was first introduced by Kondili et al. (1993) as an example and solved by their novelty general-purpose algorithm for short-term scheduling. The optimization results achieved by the author will also be used later.

While selecting the case study, it was vital that it was representative of the target industry; otherwise, results will not be relevant or representative either. The problem should have enough complexity to pose most of the expected difficulties and therefore give realistic feedback about the effectiveness of the approach used. More specifically, it was important that features and complexities, such as the following were present:

- resource allocation flexibility;
- multi-equipment tasks;
- competition between different tasks for equipment;
- and complex production sequences.

This particular case study has all of them and also introduces some others like variable batch sizes and merging and splitting of batches.

Finally, the use of a known problem also has some advantages. Bringing a new approach to a known and relevant problem, not only facilitates evaluation and comparison between most commonly used techniques but it also typically means that the results will be more useful.

3.2.1. Chemical process

The case study consists of a multipurpose batch chemical plant with three different feed stocks (FeedA, FeedB and FeedC), four intermediate products (IntAB, IntBC, HotA, and ImpureE) and two final products (Product1 and Product2).

All the tasks, task ratios, and product relationships can be seen in Figure 3.1, and task identification can be seen in Table 3.1.

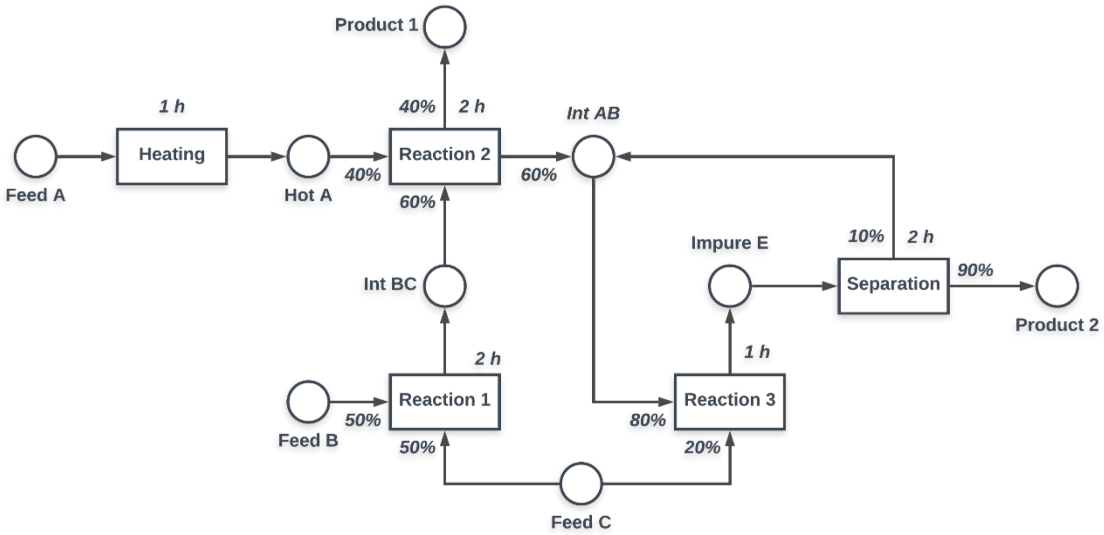


Figure 3.1. Process diagram (Kondili et al., 1993).

Task	Hearting	Reaction 1	Reaction 2	Reaction 3	Separation
Task number	1	2	3	4	5

Table 3.1. Task numeration.

3.2.2. Plant equipment and storage

The plant considered in this problem has three different types of equipment: heaters, reactors, and stills; their capacity and utility are shown in Table 3.2. Regarding storage, there are nine different storage tanks in this plant. Their corresponding material of storage and capacity are as follows:

- FeedA, FeedB, and FeedC with unlimited capacity;
- HotA with 100 Kgs of capacity;
- IntAB with 200 Kgs of capacity;
- IntBC with 150 kgs of capacity;
- ImpureE with 100 kgs of capacity;
- ProductA and ProductB with unlimited capacity.

Equipment	Capacity [Kgs]	Tasks performed
Heater	100	1
Reactor 1	80	2, 3, 4
Reactor 2	50	2, 3, 4
Still	200	5

Table 3.2. Plant equipment (Kondili et al., 1993).

3.3. Optimization model

The optimization model selected for this work will be a pivotal part of the success of this approach. As mentioned at the beginning of this chapter, detailed models are expensive, time-consuming, and highly specific; choosing such an algorithm would mitigate a lot of the advantages of discrete-event simulation. In fact, the use of a very efficient optimization algorithm could be seen as an alternative method to the one proposed in this dissertation, despite the disadvantages already pointed out.

For this reason, the optimization model used will be a deterministic general-purpose algorithm, the previously mentioned STN, the same one used by (Kondili et al., 1993). General-purpose algorithms have the advantage of being useful in a wide variety of applications, their problem modeling is quick and straightforward, but then again, it usually leads to oversimplified problems and therefore, aggregate solutions.

This model will benefit from simulation since some it is analytical simplifications can be addressed by the use of a carefully constructed simulation model, where these optimization oversights can be corrected, and it is results calibrated, allowing for the solution to move a lot closer to truly optimal results.

3.3.1. Mathematical formulation

Since both the formulation and the case study are the same ones introduced by (Kondili et al., 1993), this chapter will briefly present the mathematical formulation.

This scheduling model is based on a discrete-time system representation that has the advantage of facilitating formulation by offering a reference grid where operations are positioned against.

The variables utilized are a three-dimensional binary variable, a three-dimension real variable, and a two-dimensional real variable, explained respectively as follows:

- $W_{ijt} = 1$, if unit j starts processing task i at the beginning of time period t; 0 otherwise.
- $B_{ijt} =$ amount of material which starts undergoing task i in unit j at the beginning of time period t.
- $S_{st} =$ amount of material store in state s, at the beginning of time period t.

As for the constraints, it is fundamental that equipment allocation conflicts are resolved and that limitations in the capacity of storage and material balances are respected. To solve such problems, the following constraints were considered:

- Allocation constraints;
    - At any given time *t*, an idle equipment can only start a single task, leading to the following constraint (Kondili et al., 1993):

$$\sum_{i \epsilon Ij} W_{ijt} \leq 1$$

  - An equipment performing a task cannot start a new one until the current one is finished; this constraint can be formulated in the following way (Kondili et al., 1993):

$$\sum_{i' \in I_j} \sum_{t'=t}^{t+p_i-1} W_{i'jt'} - 1 \leq M\left(1 - W_{ijt}\right) \quad \forall j, t, i \in I_j.$$

- Capacity limitations;
  - The quantity of material that starts undergoing task *i* in unit *j* at time *t* is bound by maximum and minimum capacities of that unit (Kondili et al., 1993). This is represented by the following inequation:

$$W_{ijt}V_{ij}^{min} \leq B_{ijt} \leq W_{ijt}V_{ij}^{max} \quad \forall i, t, j \in K_i.$$

  - The quantity of material stored in a state *s* must not exceed, at any point, the maximum capacity of this state (Kondili et al., 1993). This constraint was modeled in the following way:

$$0 \leq S_{st} \leq C_s \quad \forall s, t.$$

- Material balances;
  - The net increase ($S_{st} - S_{st,t-1}$) in the quantity of material stored in a state *s* at time *t* is given by the difference of the quantity produced in this state and the quantity used and was formulated in the following way (Kondili et al., 1993).

$$S_{st} = S_{s,t-1} + \sum_{i \in T_s} \bar{\rho}_{is} \sum_{j \epsilon K_i} B_{i,j,t} - \rho_{is} - \sum_{i \in T_s} \rho_{is} \sum_{j \epsilon K_i} B_{ijt} \quad \forall s, t.$$

Finally, the objective function is presented as follows, with a few simplifications concerning the case study such as the disregard for the cost of utilities, product deliveries and raw material receipts.

$$_{max}\ profit = value\ of\ products - cost\ of\ feedstocks - cost\ of\ storage$$

The value and cost equations, in which $C_{s,H+1}$ is the cost of having material left in the system after the process has finished, C_{st} is the input price associated with the material in state *s* at time *t* and C_{st}^{s} is the running cost of storing a unit amount of material in state *s* over the interval starting at time *t*, are the following (Kondili et al., 1993):

$$value\ of\ products = \sum_{s} (\, C_{s,H+1} S_{s,H+1})$$

$$cost\ of\ feedstocks = \sum_{s} (\, C_{s0} S_{s0})$$

$$cost\ of\ storage = \sum_{s} \sum_{t=1}^{H} C_{st}^{s} S_{st}$$

3.3.2. Optimization results

The results from the STN optimization model are represented in the Gantt chart of Figure 3.2 as a production plan, where the task type is the top digit and the batch size in kgs is the bottom one, and in the intermediate storage utilization profiles for IntBC, IntAB and ImpureE are represented of Figure 3.3.

It is also important to note that even though these results were taken from the literature (Kondili et al., 1993), the Gantt chart was redone with decimal values instead of rounded ones.

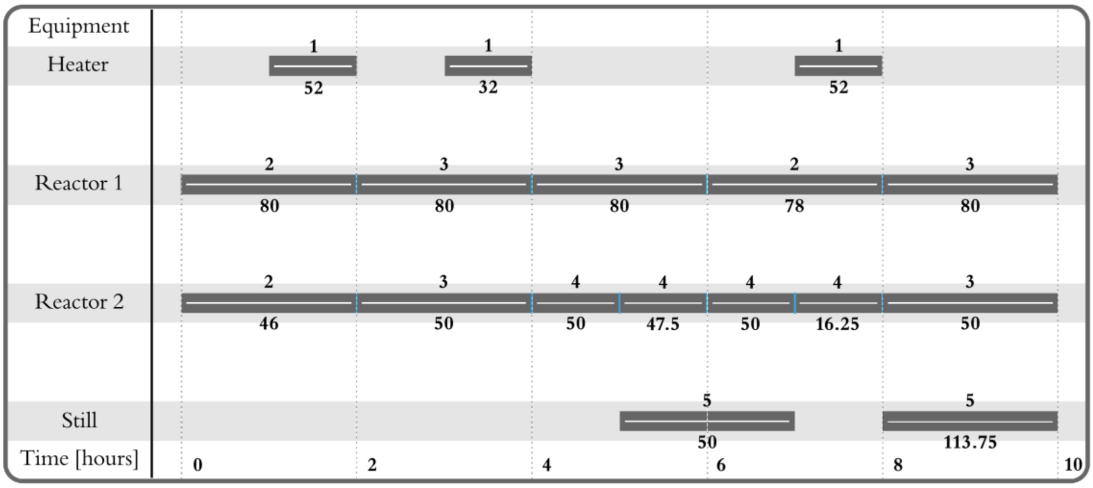


Figure 3.2. Production plan Gantt chart based on (Kondili et al., 1993).

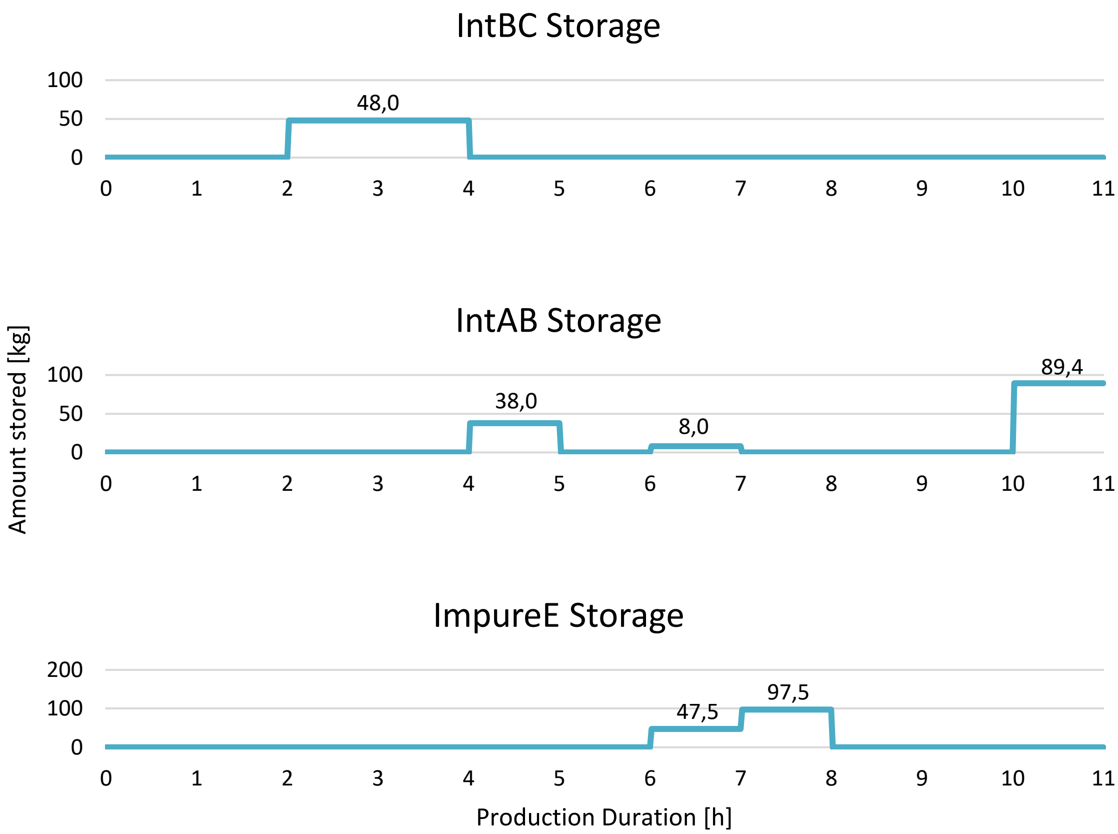


Figure 3.3. Storage utilization profiles for IntBC, IntAB and ImpureE (Kondili et al., 1993).

3.3.3. Analysis of optimization results

For a proper understanding of the flaws and opportunities of the optimization model used, it is critical to analyze its results.

Observing the Gantt chart from Figure 3.2, it is quite clear that tasks start immediately after one another. In the real world, there are filling and emptying times that can have a significant impact on the total process time.

By not considering this detail, it can also be assumed that fluid transfer logistic problems are not considered either. For example, when filling equipment with two or more different materials, it is possible to have logistic problems that force one of them to wait until the other is completely transferred. All of these details will contribute to longer process times.

Another essential aspect of a real process that is not considered in this analytic model is the process variability. In real life, no task ever takes the exact amount of time, every time. When a task is performed, there is always a slight variation in its processing time and it will also have an impact on the final process time.

Finally, we can consolidate the identified flaws as the following opportunities for improvement: i) consideration of the full pipe network, ii) modeling the process variability.

3.4. Simulation model

A well-constructed simulation model is essential for the success of this approach, more specifically, one that complements the optimization model. With that in mind, the lab team of industrial management, created a discrete-event simulation model that considers several structural aspects of the plants, as well as detailed process characteristics such as material transfer rates and material transfer logistics.

Furthermore, the design of this model was not only done considering the present case, but it was also done considering a variety of situations that the case study belongs to. The objective was to create a model that would simulate an extensive amount of different plant configurations and production plans with a minimal amount of model changes.

To achieve this goal, the simulation model was created around a modular production plan. By splitting a production plan into single sequential tasks containing all the information about the step and by creating a program that reads this information and controls the plant accordingly, it is possible to have a working model regardless of the production plan. This approach is valid only if the equipment and layout remain unchanged between production plan alterations. Still, even if one desires to use a different layout, it can be altered in the GUI with minor changes, and the model's functionality would remain.

The next chapter will go through the process of simulation package selection, and subsequent ones will describe the three main aspects of this model: the construction of the factory layout, the creation of the production plan, and the workings of the control program. The final chapter will explain how the model validation was conducted.

3.4.1. Selection of simulation software

The first step is choosing the simulation software. One can have continuous simulation for modeling continuous functions. Discrete-event simulation, on the other hand, describes systems by dividing the time frame into small intervals and only applying changes when this amount of time passes. The difference between both is comparable with the difference between an analogical signal and a digital one, respectively. The downside of describing continuous systems using discrete-event simulation, is that the results become only approximations of the actual system solution.

If this genre of problems is analyzed in regard to the type of variables that operate in it, it can be swiftly concluded that the only significant continuous variables present are the ones with respect to the material transfer. In this type of problem, the main concern when it comes to this task element is that the filling/emptying time and amount are correctly represented. By changing the representation of these values from continuous to discrete, the quantities of material inside the tanks, equipment, and pipes will no longer change continuously but in small time intervals. This means that, while a transfer is occurring, the material storage values of the equipment involved will be only approximations and so will be the transfer time. If the discrete-time intervals are very small, which they are, these approximations will not affect in any way the final results.

The specific simulation package used was SIMIO (Pegden, 2007). Its intuitive GUI, comprehensive tutorials, feature-heavy environment, and the flow library made the simulation process very clear and straightforward.

3.4.2. Construction of the plant layout

The plant layout is created through the graphical interface of SIMIO by selecting pre-existing objects. Objects are the different construction blocks offered by the software. They are meant to represent the physical equipment of a problem. The equipment arrangement is inherent to the problem at hand, so if this model is to be used in a different problem, then the layout must be rearranged or completely redone.

To create a faithful layout, objects that represent the plant's equipment must be selected, positioned, and connected in a way that represents the exact flexibility of the plant.

The first step in the creation of this layout is to select the objects that will represent each physical equipment of the plant. The next subchapter will go through the relevant objects and explain which ones will be used. Then the construction of the case study layout will be described.

3.4.2.1. Selection of SIMIO flow handling objects

SIMIO, like a few of the available discrete-event packages, is equipped with a series of features that immensely facilitate the simulation of plants involving the flow of materials. To create a simulation model, there are a number of objects available in the software that simulates real-life equipment, most of them can be programmed in a variety of ways. The chosen software has a library of objects specifically dedicated to flow handling called *flow library*, the objects used in this simulation will all come from this library. In the following paragraphs, each object's functionality will be explained; used objects will be identified as well as their utility.

To create and delete flows, *Flowsources* and *Flowsinks* are used respectively. In this simulation model, *Flowsources* are used to represent raw material storage tanks, more specifically, the ones storing FeedA, FeedB and FeedC. *Flowsinks* are not used since the analysis of the process will be simplified if all material remains in the system; instead, *Tanks* with infinite capacity will be used to store the final products.

In Figure 3.4 the visual representation of both these objects, connected by a *pipe*, can be seen.

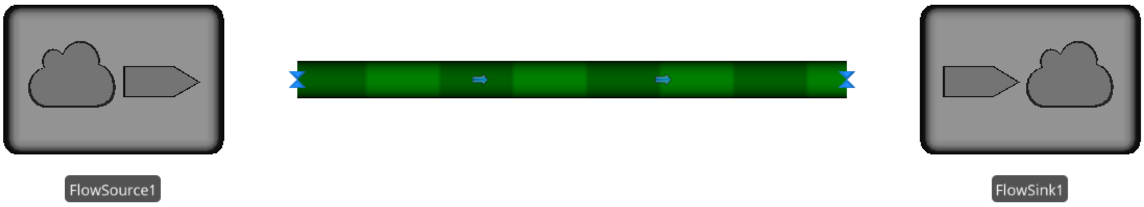


Figure 3.4. A *FlowSource* creating material flow which is then deleted by a *FlowSink*.

As for storage, SIMIO has 2 different options, *Tanks*, and *Container Entities*. The difference between them can be seen as the difference between a static storage tank and a barrel; both of them store material but one is locked in position and generally can hold a lot more and the other is a means for moving material. For our specific problem, there is no reason to use *Container Entities*, all material transfers are done through *pipes* so this type of object will not be used. *Tanks*, on the other hand, will represent both storage and equipment. In our specific case study, *Tanks* will store the products, intermediate material and will serve as reactors, heaters and stills. The equipment functionality is given by the use of processes, the SIMIO programming feature. These are later explained in 3.4.3 and 3.4.4.

The representation of Tanks and Container entities can be seen in in Figure 3.5.

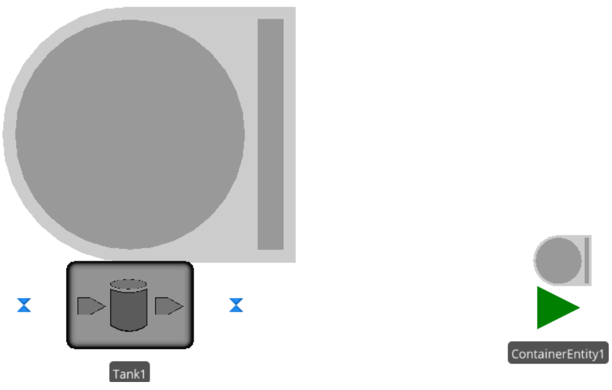


Figure 3.5. The graphical representation of a *Tank* and a *Container Entity*.

When it comes to material transfer, it can be done physically by the use of *Pipes*, *Connectors*, and *FlowNodes*.

Pipes and *Connectors* are the objects that transfer material between other objects; they can both be seen as pipes, in fact, they are identical in the simulation program, the main difference between them is that the *connector* considers both the distance between equipment and pipe volume to be null, meaning material is transferred instantly. For the simulation of the case study, the material will be transferred using *pipes* which are much more representative of the real-world transfer.

FlowNodes, regulate the flow in and out of objects or the flow at network links. These objects can be used to regulate the merge of two flow streams or the division of a single one. This object can also be used to completely obstruct flow, meaning that it can work as a valve. This will be used extensively in the creation of this simulation model.

Pipes, *Connectors* and *FlowNodes* are represented in Figure 3.6.

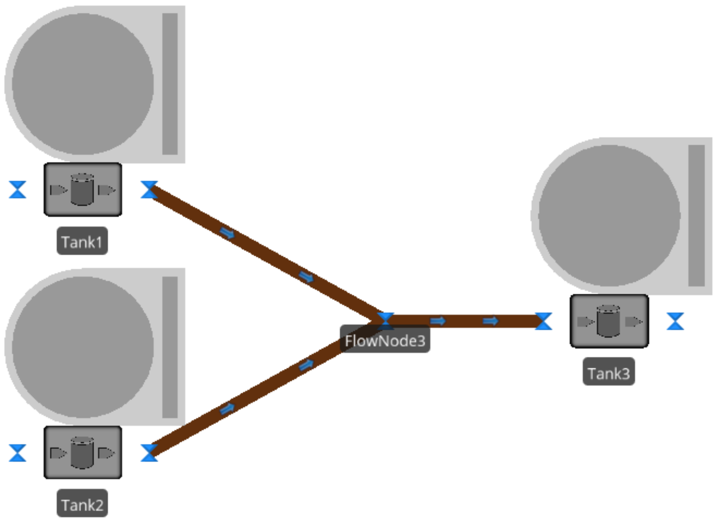


Figure 3.6. The graphical representation of *Pipes, Connectors* and *FlowNodes*.

Finally, the last two objects present in the *Flow Library* of SIMIO, are also the objects of least importance for this particular problem, the *ItemToFlowConverter*, and the *FlowToItemConverter*. They work as their name suggests, to transform flow into discrete physical items and the other does the exact opposite. They will not be used in this particular problem since it does not concern the use of physical items. One that might have the creation of a product that consumes both materials and items will benefit from these objects.

These SIMIO objects can be seen in Figure 3.7.

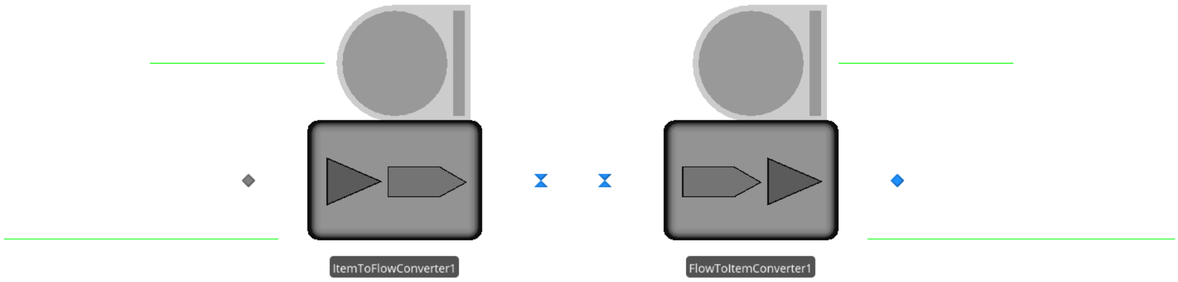


Figure 3.7. The graphical representation of *ItemToFlowConverter* and *FlowToItemConverter*.

3.4.2.2. Layout Creation

When creating the plant layout, the main objective is to have the exact amount of freedom to the factory that it would have in real life. Too much flexibility and the results will not be admissible in the real-world plant, and lack of flexibility, when compared to its real-life counterpart, would mean the results are not optimal.

For this particular case study, the layout can be seen in Figure 3.8.

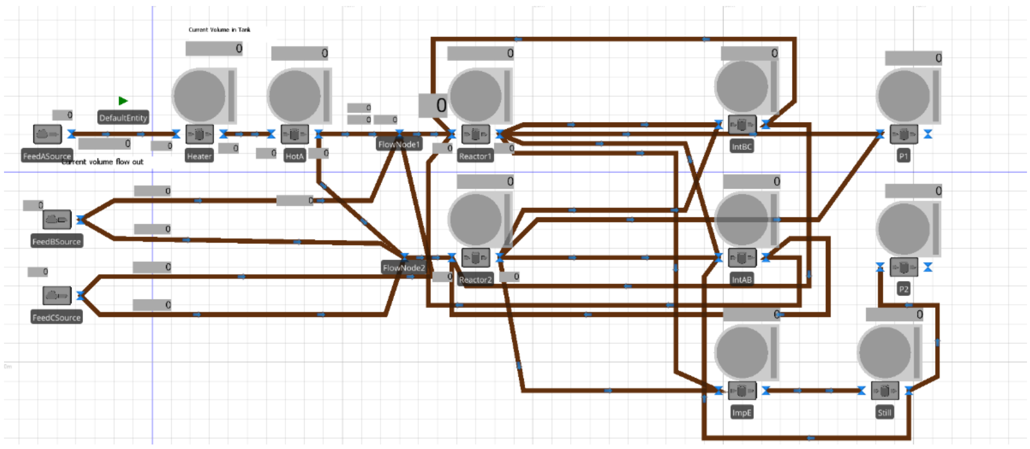


Figure 3.8. Case study plant representation in the graphical interface of SIMIO.

The first step in creating the virtual plant was to introduce the main equipment, the *FlowSources* for the different types of feeds, and the *Tanks* for both storage and equipment. After all the equipment had been placed, then *Pipe* lines and *FlowNodes* were carefully constructed to represent all the connections in the plant correctly. All the *Tanks* and *FlowSources* outputs were disabled and the correct capacity assigned. Merging and splitting *FlowNodes* were also configurated.

To finalize the construction of the plant in the GUI, all pipes were assigned a variable weight, which is controlled by the process responsible by the plant behavior. Pipe weight is a way to decide the ratio of material in a flow split, if one node is splitting the flow between two pipes and one has the weight of 0.3 and the other has the weight of 0.7, those will be the percentages by which the material will be divided.

By having variable pipe weights, it is possible to disable flow to certain pipes by setting it to zero. This feature will be very useful later in the programming part of the simulation model.

3.4.3. Development of Production plan

The main objective of the production plan is to have all the needed information available in a way that facilitates both the functioning of the model and the creation of different production plans for other problems.

This production plan will follow a sequence of different tasks that are to be performed one after the other, according to the scheduling obtained by the optimization model. Even if the optimization production plan decided that tasks should be executed simultaneously, one must go before the other in this production plan. As long as it is possible to perform the tasks simultaneously, the fact that in the simulation production plan, one is before the other does not affect the time of production. The program responsible for executing all the tasks runs in a residual amount of time and they will still be executed simultaneously.

As for the simulation production plan itself, it is implemented in a table. It could be written in Excel and later imported into the simulation software, which is advisable since Excel has a lot of features that facilitate its creation.

Table 3.3 contains all the different columns of the production plan and the information regarding each one.

Column	Type of variable	Variable	Explanation
Sequence	Integer	/	Number by which different tasks are ordered
Task	String	Task	The task that is to be performed
Material In	String	Material type	The type of material that will be consumed
Type Flow In	String	Type of object	The type of object the material is leaving
In Sto Unit	String	Object	The actual object the material is leaving
Pipe Proc In	Integer	/	Identification of the input pipe in relation to their Tank/equipment
In Sto Id	Integer	/	Identification of the origin tank/equipment
Proc Unit	String	Object	The object that will process the task
Proc Unit ID	Integer	/	Processing equipment identification
Pipe Proc Out	Integer	/	Identification of the output pipe in relation to their Tank/equipment
Out Sto Unit	String	Material type	The output storage unit.
Processing Time	Real	Hours	The number of hours a process will take
Batch	Real	Kilograms	Batch size of the task
Consumption	Real	Ratio	Percentage of the total amount to be transferred to an equipment, that the task must transfer
Production	Real	Ratio	Percentage of the total amount to be transferred out of an equipment, that the task must transfer

Start Processing time	Boolean	/	True, if the task is the final filling task for the production of the respective material, meaning the equipment can start processing. False otherwise
Flowrate	Real	Kilograms/ Hour	Flowrate by which the material is transferred during the task
Precedence	Integer	/	The number of the last task that must be performed before the considered task can be performed
Expression	String	Expression	The expression responsible to ensure precedence is respected

Table 3.3. The different production plan columns and their respective information.

3.4.4. Programming the plant behaviour

SIMIO programming, or "processes" as nominated in the software, is a type of diagrammatic programming where steps, which are different programing functions, are chained in block form. In Figure 3.9, an example SIMIO process is shown.

The flexibility and utility of this model entirely depends on its programming. It must work in a way that is general enough to be able to function regardless of the production plan and equipment layout.

This chapter will describe the different steps used and the relevant processes that control the functioning of the model.

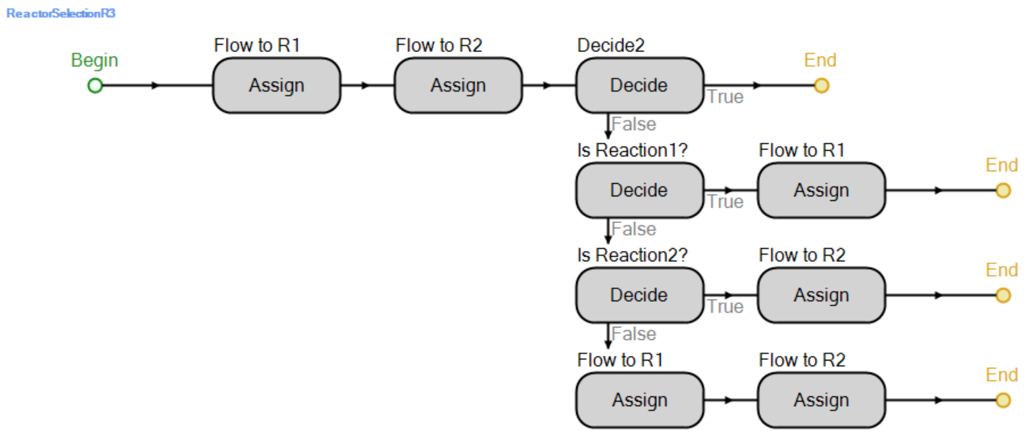


Figure 3.9. Example of a SIMIO process.

3.4.4.1. The different steps used

Within SIMIO, there are dozens of different steps to choose from, but most of them are aimed at other types of problems. For this particular plant, the used steps were those with a more general application. In the following paragraphs, all the used steps will be described and their utility explained.

Decide – This step acts similarly to an if condition in common programming languages. Given a certain condition, the function will return a true or false statement. More specifically, this step is used to check the type of task that is being performed.

Assign – This step is used to assign a new value to a specific variable. In the context of this model, it is used to open or close valves, select pipes, etc.…

Scan – The Scan step pauses the process until an assigned condition is met. In this model, one of its uses is to make sure pipes are empty before transferring material again.

Delay – This step is used to hold the process for a specific amount of time. It is used to either simulate the processing time of an equipment or to count the amount of time a valve will be open.

Search – This step verifies if an object, table, or condition has changed from its original value. It is used to check if the process has just started or if it has been running already.

Execute – It is used to execute a specific process and is used to start the main process after the right conditions are met.

Notify – It is used to notify the user if a condition was met. This step is very useful for debugging and testing. It is not necessary to the process's functionality.

3.4.4.2. Process functioning

As previously mentioned, the whole plant functioning is created through a process which takes its information from a production plan. This production plan is divided into a set of sequential production plan tasks; for the sake of simplicity, they will be referred to as PPTs. The distinction between PPTs will be important later on in the explanation of the main process. Figure 3.10 shows the way PPTs are classified.

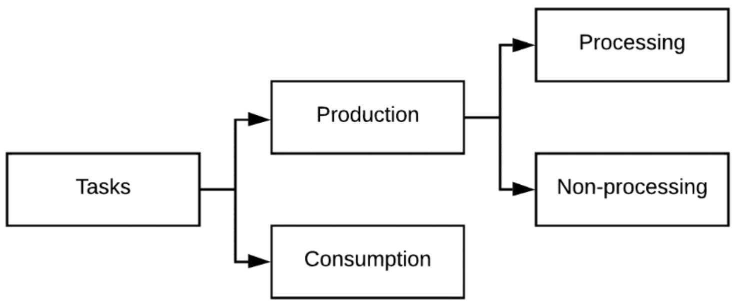


Figure 3.10. PPTs sub-types.

PPTs are divided into these categories because, due to the way that the process responsible for the functioning of the plant was created, PTTs can only input one type of material. The same goes for output, but it does not have to be the same material as input, for obvious reasons.

This means that for the execution of a task that will have two different material proceeds like Reaction2 or Separation, the first PPT can only remove one of the materials from the equipment. Consumption PPTs are used to remove the second material from the equipment.

Processing and non-processing PPTs exist for the exact same reason but related to input. When tasks require two different types of material to be performed, two PTTs must be used to fill the equipment and only the second one should be of the processing type. The reason behind it is quite simple, the model needs to know when to kickstart the process, since the tank will be full only after the second PPT is performed, that is when it should start processing. As an example, when performing Reaction2 three different PPTs are required, first a non-processing PPT to fill the reactor with the first material, then a processing PPT to finish filling the reactor, perform the reaction and remove the first product from the equipment and finally a consumption PPT to remove the second product.

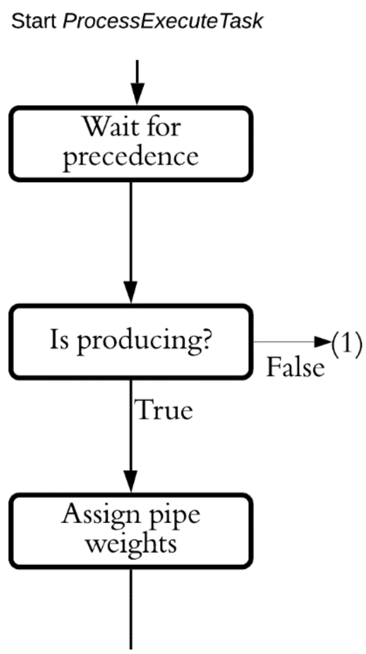
There are only three different processes that operate this simulation model, *ProcessExecuteTask* which is the main process in charge of the model's operation and also the main focus of this chapter, *ProcessSchedul*e which is in charge of starting the main process and to keep track of the PPT number and *ProcessStock* which is used to record tank and equipment stocks for later analysis.

The rest of this chapter will be dedicated to the functioning of *ProcessExecuteTask* and its explanation will follow Figure 3.11.

The process is initiated by *ProcessSchedule*. This simple process starts running as soon as the model does, and it is responsible for initiating production and finalizing it when the production plan reaches its end. It is also the process that keeps track of the tasks performed.

After *ProcessExecuteTask* is initiated, it verifies if the current PPT can be performed by making sure the one immediately before was performed successfully. After this condition becomes positive, the process moves on.

A decision type step is performed next, the program verifies if the PPT is of the production type. If true the process moves on; otherwise the process skips ahead (1). Then, an assign step assigns the correct pipe to transfer the material from the output to the input object.



Next, the process reaches another decide step, which is used to verify the type of object from which the material will leave. This must be done because, for the same purpose, expressions vary depending on the object. So, even though the next steps are exactly the same either way, the distinction must be made because each requires different syntax. This type of step is used more than once throughout the process but it will be omitted from now on.

Then, the scan step waits for both the output valve of the output object to be closed and for any material left in the queue to enter the input object. Finally, when the conditions are met, the process moves to an assign step that opens the valve from the output object and fills the target equipment.

The time the output valve is opened is calculated using the transfer rate and the batch size and the consumption ratio, so it outputs the exact amount of material. This is done through a delay step.

The next step closes the output valve immediately after the calculated amount of time has passed, followed by a scan step that waits until all the material has been transferred. The subsequent step is of the decision type once again, checking whether the task is of the processing or the non-processing type. If the task is of the non-processing kind, the process ends and the model starts with the next PPT on the sequence. If the PPT is of the processing type, the process moves on to a delay step which will last for the full processing time of the task. When the time is over, the equipment is finished processing and the products are ready to be removed. A scan step waits until the pipes leading to the input tank are free, then the process moves to an assign step that opens the output valves of the equipment followed by a delay step that calculates the time for the material transfer. The process ends with an assign step closing the output valves of the equipment and by restarting once again on the next PPT.

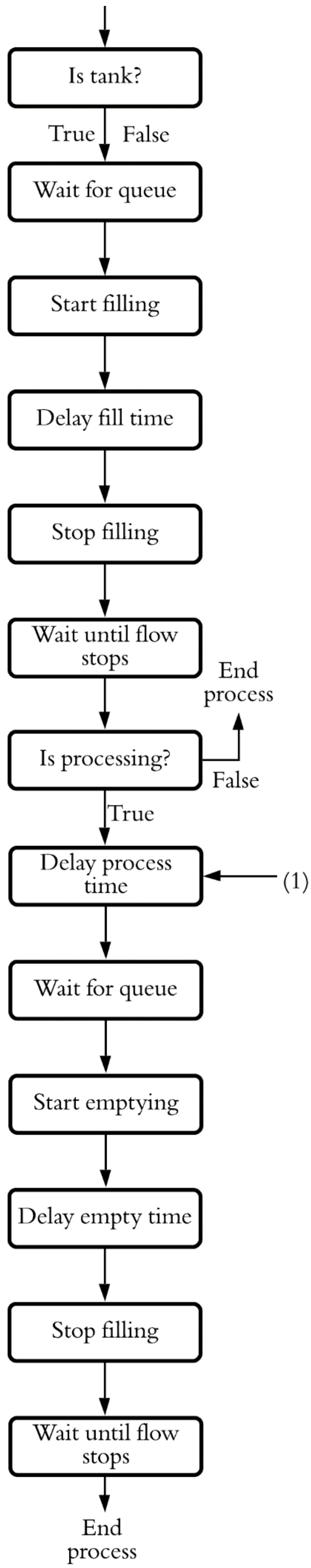


Figure 3.11. Representation of *ProcessExecuteTask*.

3.4.5. Validation of the model

Before the simulation model is concluded, it is important to make sure it actually functions correctly and faithfully represents the problem at hand. To verify the validity of the model one main objective test was conducted. The test consists of reproducing the problem representation of the analytical model in the simulation model and comparing the analytic results with the simulation results. To do so, material transfer-rates were accelerated, turning emptying and filling times to approximately 0 seconds.

After the model was run, the storage levels throughout time, the final product and intermediary levels and the total run time were compared to those reported in the literature.

The following graphs show intermediate storage values, in Kgs throughout 11 hours of process time, from the literature and from the simulation model with the validation parameters.

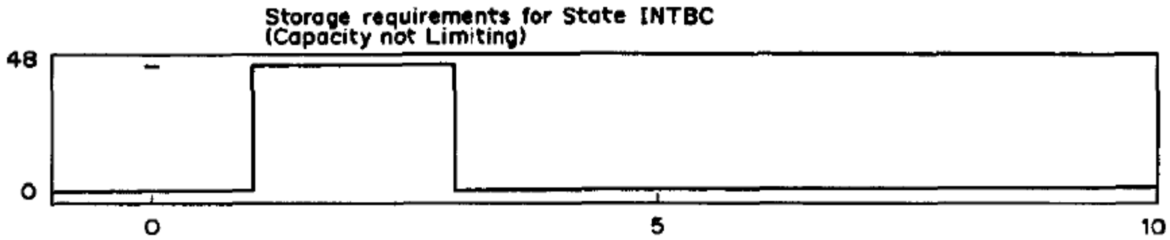


Figure 3.12. IntBC storage utilization profile from literature (Kondili et al., 1993).

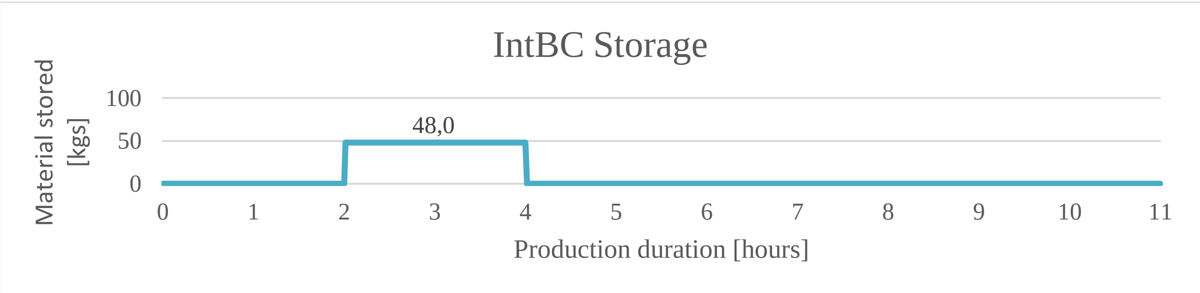


Figure 3.13. IntBC storage utilization profile according to the simulation model.

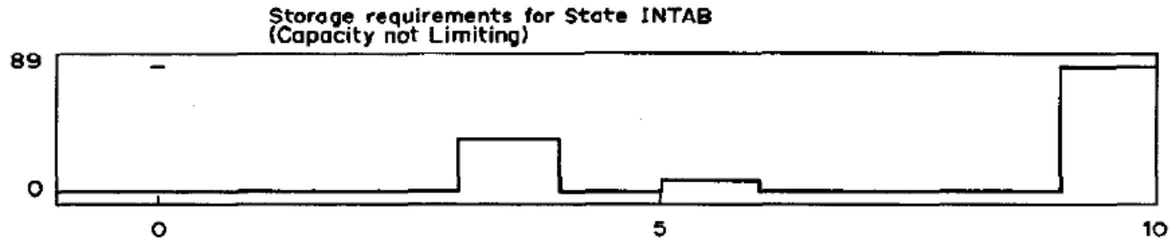


Figure 3.14. IntAB storage utilization profile from literature (Kondili et al., 1993).

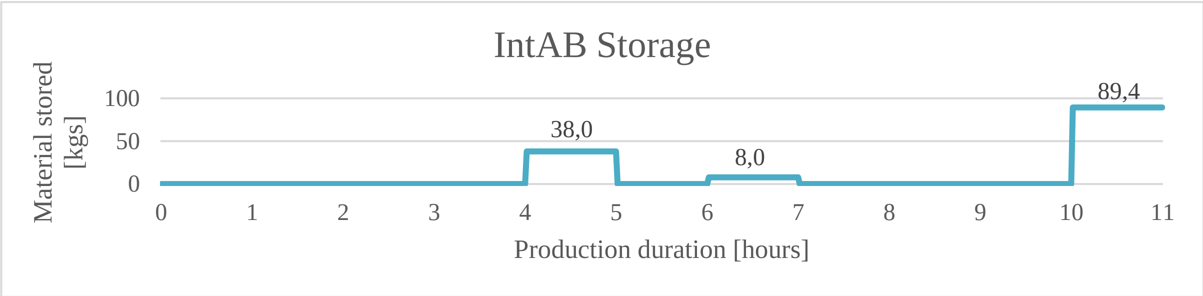


Figure 3.15. IntAB storage utilization profile according to the simulation model.

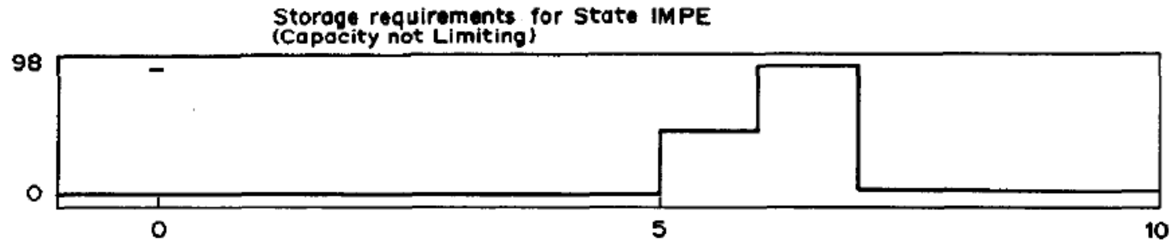


Figure 3.16. ImpureE storage utilization profile from literature (Kondili et al., 1993).

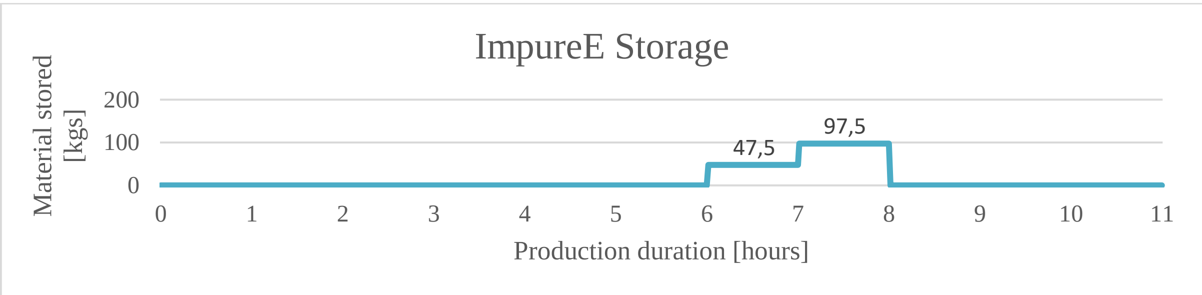


Figure 3.17. ImpureE storage utilization profile according to the simulation model.

The storage levels throughout the full production time between the two sources are exactly the same. As for the final product quantities and full production duration, the results were exactly the same as the values mentioned in the literature. The product storage levels and the production duration can be seen in Figure 3.18.

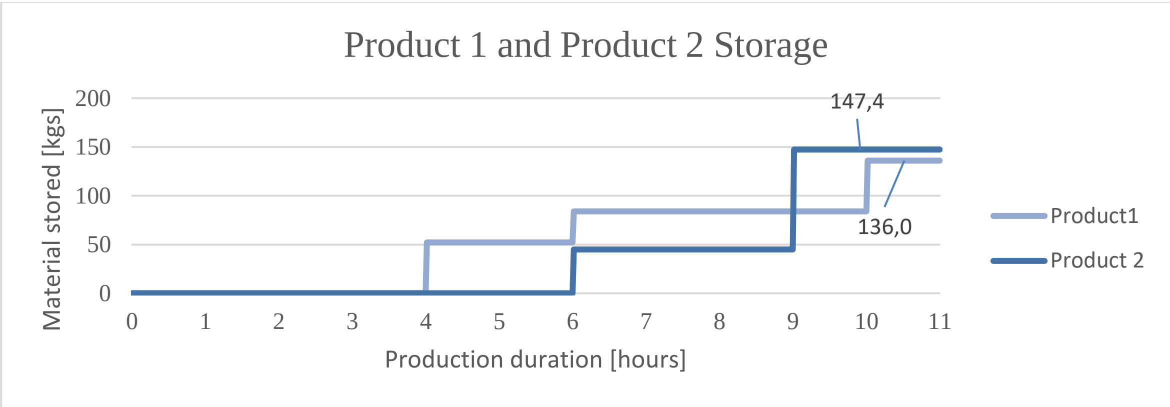


Figure 3.18. Product1 and Product2 storage utilization profile according to the simulation model.

4. APPROACH ASSESSMENT

Throughout this chapter, the solutions obtained from the previously described deterministic analytic model will be evaluated. This evaluation will now take into consideration material transfer rates and material transfer logistics, thanks to the use of discrete-event simulation. The main areas of effect of the previously mentioned features will be separated in the three subchapters: i) computational performance, where the processing speed of the model will be accessed; ii) tank sizing, where the intermediate tank sizing done by the optimization methodology will be reviewed; and iii) production duration estimates, where the estimates of the full production duration will be analyzed against the ones achieved through the simulation model.

The following aspects of the problem and plant characteristics were kept constant throughout all the calculations in this chapter:

- Pipe length of 5 meters
- Elliptical pipe cross-section, 0,25m by 0,25m (default)
- $\rho_{material} = 1000\ {}^{kg}/_{m^3}$

4.1. Computational performance

Computational performance is critical when it comes to any kind of optimization methodology. Results must be obtained in a useful time interval; otherwise, they might render useless. This work is dedicated to short-term scheduling, in which the objective is typically to make quick production decisions with the available information. If the optimization methodology used cannot find a result in the available time frame, then it is not fit to make those decisions.

Within the scope of this dissertation, only the simulation model can be evaluated for computational performance since the analytic results were taken straight from the literature (Kondili et al., 1993). Nevertheless, it is known that the analytic model considered in this scientific article is quite efficient.

To determine the computational processing duration, different production intervals were considered by repeating the production plan. The following table presents values obtained from 10 replications of each amount of production plan repetitions and considering a 500 kgs/h transfer rate as well as 95% confidence level. Regarding the CPU, all the values were calculated with an Intel i7-8750H (mobile version) and each core ran at an average 2.6GHz.

Production Plan repetitions	Average processing time [s]	Half-Width [s]	Margin of error [%]
1	0,21	0,03	14,93
2	0,79	0,07	9,24
5	3,91	0,22	5,71
10	14,22	0,3	2,13
20	57,26	1,13	1,97
30	109,42	1,07	0,98
40	233,12	6,09	2,61
50	346,61	22,32	6,44
60	468,08	46,06	9,84

Table 4.1. The processing duration, half-width, and margin of error throughout different production intervals after 10 replications.

With the objective of achieving a margin of error lower than 5%, the number of necessary replications is calculated with the values that achieved the highest margin of error. These values were found when performing the scenario in which the production plan was not repeated. The number of necessary replications was calculated in the following manner:

$$n = n_0 \frac{h_0^2}{h^2} \quad \leq> \quad n = n_0 \frac{h_0^2}{(\bar{x} * 0{,}05)^2} \quad \leq>$$

$$\leq> n = 10 \frac{0{,}03^2}{(0{,}21 * 0{,}05)^2} = 81{,}63 => 82$$

Where:

- n = Necessary number of replications to achieve the desired margin of error.
- n_0 = number of initial replications.
- h_0 = Half-width calculated from the initial set of replications.
- h = Target half-width.

Having calculated the necessary number of replications to archive the desired confidence interval, the different scenarios were run again with the calculated number of replications. The results can be seen in Table 4.2.

Production Plan repetitions	Average processing time [s]	Half Width [s]	Margin of error [%]	PR [h/sec]
1	0,26	0,01	3,85	38,46
2	0,78	0,02	2,56	25,64
5	3,83	0,08	2,09	13,05
10	14,47	0,25	1,73	6,91
20	60,10	0,87	1,45	3,33
30	135,68	1,58	1,16	2,21
40	240,68	3,03	1,26	1,66
50	368,47	4,5	1,22	1,36
60	523,13	8,36	1,6	1,15

Table 4.2. The processing duration, half-width, margin of error and PR throughout different production intervals after 82 replications.

While, for one production plan repetition, the margin of error was close to 5%, all the other values were below 3,85 %, showing that the previous replication calculation was conservative. As for computational performance, the model proved very efficient regarding the case study, simulating it's 10-hour production plan in under 1 second. Furthermore, SIMIO(Pegden, 2007) computes one replication per CPU core. This means that while one replication might take 130 seconds, in a 6 core CPU, 6 replications will take the same amount of time because they are computed in a parallel way.

However, with the increase in simulation time, the PR (processing ratio $\left[\frac{hour_{simulation}}{second_{processing}}\right]$) drops significantly. Meaning that the relationship between processing time and simulation time is not linear. This can make longer simulations of more complex processes and plants problematic. In Figure 4.1 evolution of the computational processing time needed for an increasing simulation time.

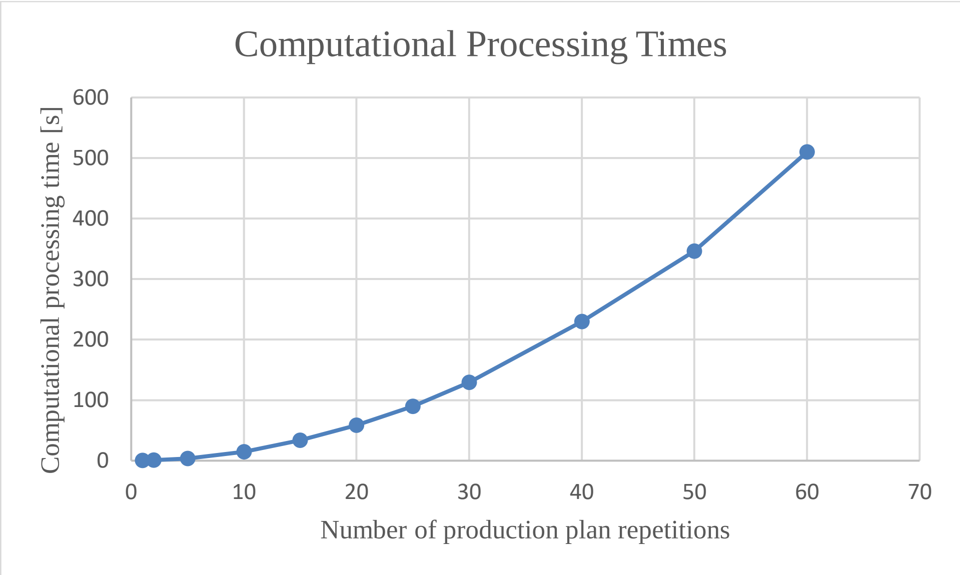


Figure 4.1. Computational processing times of different production lengths.

4.2. Tank sizing

When it comes to sizing the different components of a chemical plant, equipment capacity planning appears to be a fundamental mater. However, altering the equipment size could alter optimization results. For this reason, choosing the correct capacity is a complex matter.

Intermediate material storage, on the other hand, is directly tied to the production plan, which makes it easy to find a preliminary maximum values through the optimization methodology. The preliminary results, though, are only as accurate as the model that obtained them. If the analytic model cannot closely describe the system it is optimizing, any unconsidered system characteristics might alter the obtained results. In this chapter, we analyze the use of discrete-event simulation as mean to obtain more accurate storage intermediate profiles by introducing system characteristics the optimization model does not consider.

To do so, the maximum intermediate tank weights, throughout the full process, of the three main intermediate materials (IntBC, IntAB and ImpureE) are compared, see Table 4.3.

Tank	**50 [kgs/h]**	**500 [kgs/h]**	**5000 [kgs/h]**	**50000 [kgs/h]**	**Instant Transfer**
IntBC	96	96	96	96	48
IntAB	89.4	89.4	89.4	89.4	89.4
ImpureE	114	114	114	114	97.5
Full process duration	28:10:02	11:49:23	10:10:54	10:01:05	10:00:00

Table 4.3. Maximum tank weight [kgs] according to different transfer rates.

On this table, the maximum tank weight range throughout the full production duration can be seen through 5 different transfer rate levels of magnitude. The “Instant Transfer” values were obtained from the optimization methodology, while the rest was obtained by the simulation model.

Comparing the simulation values with the ones calculated by the analytic model, there is a clear difference. Both the tanks responsible for IntBC and ImpureE storage would be incorrectly sized through the data obtained from the analytic model. To further investigate the reason behind this change, the profile levels from the tank responsible for IntBC storage are compared at 500 kgs/h and instant transfer rate, in Figure 4.2.

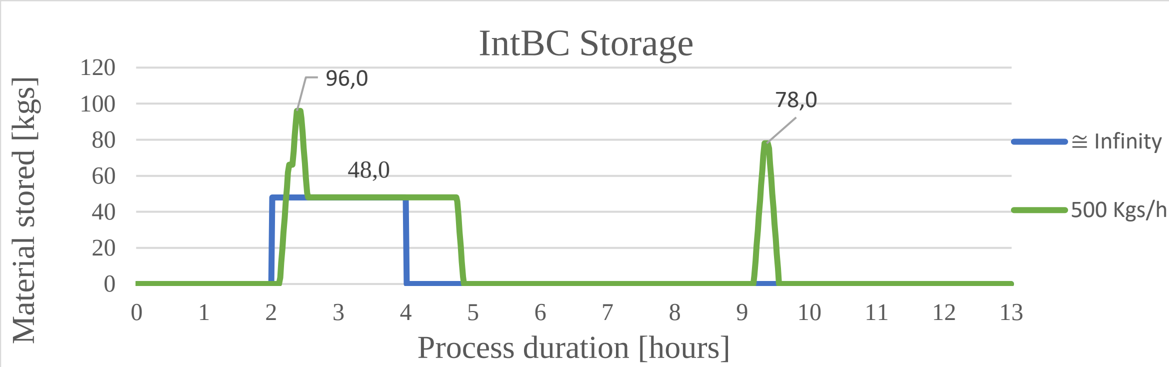


Figure 4.2. IntBC Storage utilization profiles with different material transfer rates.

This tank is the one where the most significant change in maximum weight usage difference occurs, it doubles from 48 kgs to 96 kgs. While the change is very significant, the reasons behind it are the same as the ones behind all the other changes in maximum weight.

When the instant transfer is considered, if the proceedings from a reaction are directly used in the next one, they are transferred back to the same reactor when the task is over, without the model considering them to be stored in the intermediate tanks beforehand. This leads to serious inaccuracies in the intermediate tank weight profiles. Moreover, the storage profiles of the same tank at drastically different transfer rates are exactly the same, even at extremely fast transfer rates. This seems to be a result of the phenomenon explained above since considering even the slightest filling time would cause all the material to be stored in the intermediate tanks before it is transferred back to the equipment.

This analytic simplification altered the storage profiles of all the intermediate materials, except for HotA because of its very simple logistics. Even the tank responsible for intermediate storage of IntAB, which has a constant maximum weight throughout the different transfer rates, has an immensely different storage profile from the one obtained through the optimization methodology. Figure 4.3 compares the weight profiles of considering instant transfer and 500 kgs/h.

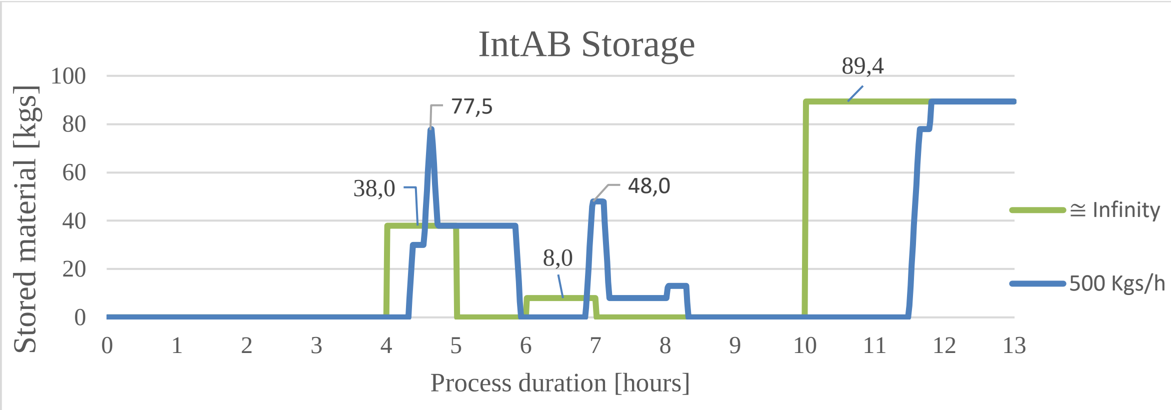


Figure 4.3. IntAB Storage utilization profiles with different material transfer rates.

4.3. Production duration estimates

The capability of correctly predicting the time a process will take to complete is important to any company that wishes to be competitive. Being able to minimize storage costs relies on correctly scheduling raw material orders and product deliveries. For example, the inaptitude to correctly order a batch of raw materials might lead to an early delivery at a time the plant has no capacity to store all of it, forcing a reschedule that might come too late and force the plant to stop production until it arrives. This type of mistake can be detrimental to a company. Moreover, if a company wishes to uphold its reputability and keep its costumers satisfied, it's imperative that it can honor its commitments by providing their products on time.

What makes this problematic, as previously stated, is that analytic methods simplifications might change the validity of their duration estimates. In this chapter, the process duration estimates obtained through the optimization method will be contrasted with the ones from the simulation model throughout different transfer rates and production horizons, see Table 4.4.

Production Plan repetitions	500 [kgs/h]	1500 [kgs/h]	2500 [kgs/h]	3500 [kgs/h]	Instant Transfer
1	11:49:23	10:36:27	10:21:52	10:15:37	10:00:00
5	59:11:48	53:02:16	51:49:21	51:28:06	50:00:00
15	177:20:23	159:06:47	155:28:04	153:54:20	150:00:00
% of total processing time	118,2%	106,0%	103,6%	102,6%	100%

Table 4.4. Process duration at different transfer rates and throughout different production intervals.

An important observation is the fact the percentage of total processing time only remains constant for the same transfer rate because the production times were calculated using repetitions of the same production plan. It would generally have slight variations if it was calculated throughout a single one. Nevertheless, this simplification facilitates observation.

The results from Table 4.4 show that logistical delay has a significant impact on the total production duration. For a transfer rate of 500 kgs/h, logistical delays amount to an almost increase 20%. Furthermore, considering the simplifications made and the results obtained, the logistical delay appears to be inversely proportional to the transfer rate. This can is better observed in Figure 4.4.

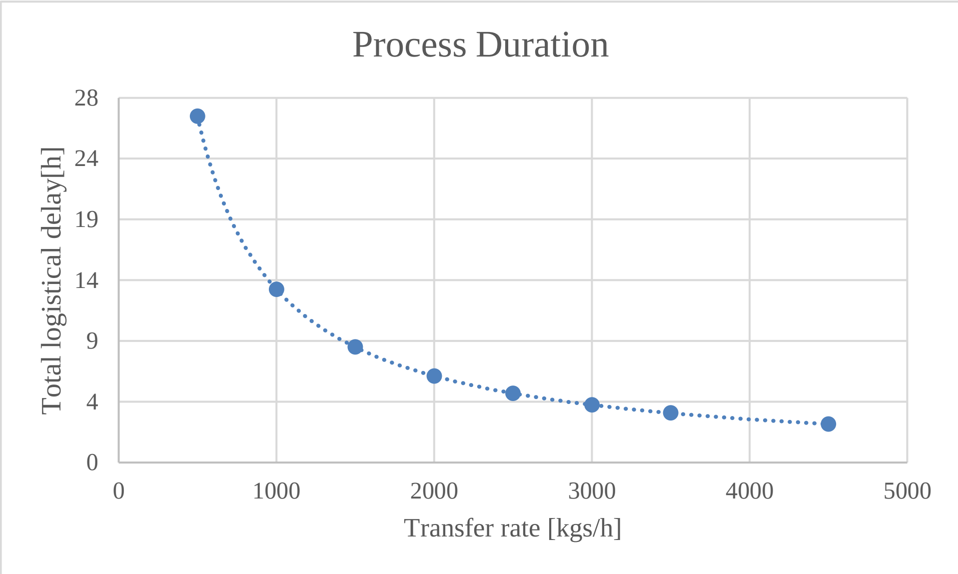


Figure 4.4. Changes in logistical delay along different transfer rates.

Knowing that the logistical delays include delays due to pipe length and pipe equipment occupation that do not scale the same way with transfer rate, it would be wrong to assume this relation to be an absolute truth. Nonetheless, logistical delays are so close to being inversely proportional because the emptying and filling times are, and since the other delays are minimal, they barely have an impact on the final values. Meaning that, for engineering purposes, this relation could be assumed for a fast interpolation and achieve quite accurate values.

It's important to note, that without the consideration of variability, there is no assurance that even the estimates obtained through simulation are close to the real production duration time.

5. CONCLUSIONS

Process optimization is an integral part of innovation, and of keeping a competitive advantage. Optimization can be a great tool in the pursuit of competitiveness. Nonetheless, it struggles with computational performance making it very difficult to optimize complex problems in useful temporal horizons.

Optimization models can be improved to achieve the needed level of performance but, most of the time, only through a long and arduous process that might not be beneficial for a lot of situations. This dissertation focusses on the use of discrete-event simulation, along with a simple optimization algorithm, as a solution for the latter. Known for being very efficient, relative to computational performance and variability evaluation, simulation has the potential to minimize the burden of analytic models drastically.

To investigate this approach, a simple MILP model and a detailed discrete-event simulation model of the manufacturing system were used to optimize a complex chemical plant following a simplified SCS methodology. Furthermore, the model was formulated in a recipe-oriented way that allows for the simulation of a wide variety of different batch chemical processes with minor model adjustment.

Since simulation's use is not to optimize but to evaluate optimization results, the analytic methodology was vital in the scheduling of optimized production plans and determining production lots.

As for the simulation methodology, the performance of the model proved to be as computationally effective as expected. A high number of long replications runs of a complex process were swiftly performed. However, PR declines with the increase in simulation time, making longer simulation of more complex processes and plants potentially problematic.

Concerning total production duration, simulation proved to be a valuable evaluation tool. The simulation model estimated total production times that were more than 10% longer than the ones estimated by the optimization model, considering realistic material transfer rates and without variability.

Furthermore, the simulation methodology was able to greatly improve the intermediate storage profiles by obtaining much more accurate maximum tank weights and tank weight profiles. These estimates would much more adequate to be used in the sizing of intermediate storage tanks.

Overall, the proposed methodology proved useful and competent in a number of aspects and delivered most of the expected benefits. Still, since this problem was taken from the literature, it’s hard to know whether or not the estimates are as accurate as they appear. There is always the possibility that some unconsidered system characteristics would impact these estimates in unexpected ways. Also, the consideration of process variability would have potentially impacted a number of these results.

5.1. Further research

Even though this methodology proved very competent, there are a number of opportunities that can improve it a lot.

The most significant of all would be a more robust implementation of variability since, as it currently stands, the formulated model struggles to implement it. Furthermore, the implementation of an iterative component, that could adjust the objective function according to the simulation results, would have the potential to achieve better scheduling and production plan. The use of a different optimization methodology such as ROSA or even the creation of a dedicated one, also has the potential to improve these results.

To further investigate the usefulness of the simulation, there opportunities to explore. The consideration of different problems could find other advantages or downsides to the methodology. Similarly, the implementation of other system dynamics has the same potential.

BIBLIOGRAPHY

Ackermann, S., Fumero, Y., & M. Montagna, J. (2018). Optimization Framework for the Simultaneous Batching and Scheduling of Multisite Production Environments. *Industrial & Engineering Chemistry Research*, *57*(48), 16395–16406. https://doi.org/10.1021/acs.iecr.8b03140

Amaran, S., Sahinidis, N. V, Sharda, B., & Bury, S. J. (2016). Simulation optimization: a review of algorithms and applications. *Annals of Operations Research*, *240*(1), 351–380. https://doi.org/10.1007/s10479-015-2019-x

Azzaro-Pantel, C., Bernal-Haro, L., Baudet, P., Domenech, S., & Pibouleau, L. (1998). A two-stage methodology for short-term batch plant scheduling: discrete-event simulation and genetic algorithm. *Computers & Chemical Engineering*, *22*(10), 1461–1481. https://doi.org/10.1016/S0098-1354(98)80033-1

Banks, J., Carson, J. S. I., Nelson, B. L., & Nicol, D. M. (2013). *Discrete-Event System Simulation* (5th intern). Pearson.

Barbosa-Póvoa, A. P., & Macchietto, S. (1994). Detailed design of multipurpose batch plants. *Computers and Chemical Engineering*, *18*(11–12), 1013–1042. https://doi.org/10.1016/0098-1354(94)E0015-F

Barton, P. I., & Pantelides, C. C. (1994). Modeling of combined discrete/continuous processes. *AIChE Journal*, *40*(6), 966–979. https://doi.org/10.1002/aic.690400608

Beis, D. A., Loucopoulos, P., Pyrgiotis, Y., & Zografos, K. G. (2006). PLATO Helps Athens Win Gold: Olympic Games Knowledge Modeling for Organizational Change and Resource Management. *INFORMS Journal on Applied Analytics*, *36*(1), 26–42. https://doi.org/10.1287/inte.1060.0189

Byrne, M. ., & Bakir, M. . (1999). Production planning using a hybrid simulation – analytical approach. *International Journal of Production Economics*, *59*(1–3), 305–311. https://doi.org/10.1016/S0925-5273(98)00104-2

Chen, Y., Mockus, L., Orcun, S., & Reklaitis, G. V. (2012). Simulation-optimization approach to clinical trial supply chain management with demand scenario forecast. *Computers & Chemical Engineering*, *40*, 82–96.

https://doi.org/10.1016/J.COMPCHEMENG.2012.01.007

Crooks C. (1992). *Synthesis of operating procedures for chemical plants* (Vol. 1). University of London.

Czulek, A. J. (1988). An experimental simulator for batch chemical processes. *Computers & Chemical Engineering*, *12*(2–3), 253–259. https://doi.org/10.1016/0098-1354(88)85036-1

Engell, S., Fritz, M., Schulz, C., & Wöllhaf, K. (1995). Hybrid Simulation of Flexible Batch Plants. *IFAC Proceedings Volumes*, *28*(9), 123–128. https://doi.org/10.1016/s1474-6670(17)47027-3

Fahrland, D. A. (1970). Combined discrete event continuous systems simulation. *SIMULATION*, *14*(2), 61–72. https://doi.org/10.1177/003754977001400203

Figueira, G., & Almada-Lobo, B. (2014). Hybrid simulation–optimization methods: A taxonomy and discussion. *Simulation Modelling Practice and Theory, 46*, 118–134. https://doi.org/10.1016/J.SIMPAT.2014.03.007

Fruit, W. M., Reklaitis, G. V., & Woods, J. M. (1974). Simulation of multiproduct batch chemical processes. *The Chemical Engineering Journal*, *8*(3), 199–211. https://doi.org/10.1016/0300-9467(74)85025-2

Fu, M. C. (1994). Optimization via simulation: A review. *Annals of Operations Research*, *53*(1), 199–247. https://doi.org/10.1007/BF02136830

Fu, M. C. (2002). Feature Article: Optimization for simulation: Theory vs. Practice. *INFORMS Journal on Computing, 14*(3), 192–215. https://doi.org/10.1287/ijoc.14.3.192.113

Griva, I., Nash, S. G., & Sofer, A. (2009). *Linear and Nonlinear Optimization: Second Edition*. Retrieved from https://books.google.hu/books?id=uOJ-Vg1BnKgC

Joglekar, G. S., & Reklaitis, G. V. (1984). A simulator for batch and semiconductors processes. *Computers & Chemical Engineering*, *8*(6), 315–327. https://doi.org/10.1016/0098-1354(84)90001-2

Jung, J. Y., Blau, G., Pekny, J. F., Reklaitis, G. V., & Eversdyk, D. (2004). A simulation based optimization approach to supply chain management under demand uncertainty. *Computers & Chemical Engineering*, *28*(10), 2087–2106. https://doi.org/10.1016/J.COMPCHEMENG.2004.06.006

Kondili, E., Pantelides, C. C., & Sargent, R. W. H. (1993). A general algorithm for short-term scheduling of batch operations—I. MILP formulation. *Computers & Chemical Engineering*, *17*(2), 211–227. https://doi.org/10.1016/0098-1354(93)80015-F

Law, A. M. (2013). Simulation Modeling and Analysis. In *Simulation Modeling and Analysis* (Fifth Edit). Retrieved from www.averill-law.com

Lee, H., & Maravelias, C. T. (2017). Discrete-time mixed-integer programming models for short-term scheduling in multipurpose environments. *Computers & Chemical Engineering*, *107*, 171–183. https://doi.org/10.1016/J.COMPCHEMENG.2017.06.013

Marques, C. M., Moniz, S., de Sousa, J. P., & Barbosa-Póvoa, A. P. (2016). Optimization and Monte Carlo Simulation for Product Launch Planning under Uncertainty. *Computer Aided Chemical Engineering*, *38*, 421–426. https://doi.org/10.1016/B978-0-444-63428-3.50075-8

Marques, C. M., Moniz, S., de Sousa, J. P., & Barbosa-Póvoa, A. P. (2017). A simulation-optimization approach to integrate process design and planning decisions under technical and market uncertainties: A case from the chemical-pharmaceutical industry. *Computers & Chemical Engineering*, *106*, 796–813. https://doi.org/10.1016/J.COMPCHEMENG.2017.04.008

Mauderli, A., & Rippin, D. W. T. (1979). PRODUCTION PLANNING AND SCHEDULING FOR MULTI-PURPOSE BATCH CHEMICAL PLANTS. *Journal of Microwave Power*, *1*(1–4), 348–360. https://doi.org/10.1016/0098-1354(79)80033-2

Nocedal, J., & Wright, S. (2006). *Numerical Optimization*. Retrieved from https://books.google.co.in/books?id=VbHYoSyelFcC

Pantelides. (1994). Unified Frameworks for optimal process plannning and scheduling. *Proceedings on the Second Conference on Foundations of Computer Aided Operations*, 253–274. Cache Publications New York.

Papageorgiou, L. G. (2009). Supply chain optimisation for the process industries: Advances and opportunities. *Computers & Chemical Engineering*, *33*(12), 1931–1938. https://doi.org/10.1016/J.COMPCHEMENG.2009.06.014

Pegden, C. D. (2007). *SIMIO*. Retrieved from https://www.simio.com/index.php

Pinto, T., Barbósa-Póvoa, A. P. F. D., & Novais, A. Q. (2008). Design of multipurpose batch plants: A comparative analysis between the STN, m-STN, and RTN representations and formulations. *Industrial and Engineering Chemistry Research*, *47*(16), 6025–6044. https://doi.org/10.1021/ie071281n

Rippin, D. W. T. (1993). Batch process systems engineering: A retrospective and prospective review. *Computers & Chemical Engineering*, *17*, S1–S13. https://doi.org/10.1016/0098-1354(93)80201-W

Samuel Moniz, Ana Paula Barbosa-Póvoa, J. P. de S. (2013). New General Discrete-Time Scheduling Model for Multipurpose Batch Plants. *Industrial & Engineering Chemistry Research*. https://doi.org/10.1021/ie4021073

Seid, E. R., & Majozi, T. (2014). Heat integration in multipurpose batch plants using a robust scheduling framework. *Energy*, *71*, 302–320. https://doi.org/10.1016/J.ENERGY.2014.04.058

Shah, N. K., & Ierapetritou, M. G. (2012). Integrated production planning and scheduling optimization of multisite, multiproduct process industry. *Computers & Chemical Engineering*, *37*, 214–226. https://doi.org/10.1016/J.COMPCHEMENG.2011.08.007

Sinnott, R., & Towler, G. (2019). *Chemical engineering design: SI Edition*. Butterworth-Heinemann.

Vieira, M., Barbosa-Póvoa, A. P., Moniz, S., & Pinto-Varela, T. (2018). Simulation-Optimization Approach for the Decision-Support on the Planning and Scheduling of Automated Assembly Lines. *2018 13th APCA International Conference on Automatic Control and Soft Computing (CONTROLO)*, 265–269. https://doi.org/10.1109/CONTROLO.2018.8514297

Youle, P. V. (1960). Simulation of Full-scale Multi-stage Batchwise Chemical Plant. *The Computer Journal*, *3*(3), 150–157. https://doi.org/10.1093/comjnl/3.3.150

Zottolo, M., Williams, E. J., & Ulgen, O. M. (2007). Simulation implements demand-driven workforce scheduler for service industry. *2007 Winter Simulation Conference*, 219–225. https://doi.org/10.1109/WSC.2007.4419604