

https://dspace.org/

Synthesis and Optimization of a Water Network Scheme for a Multipurpose Multiproduct Textile Dyeing Facility: An Industrial Case Study

Tsheoga, Kamogelo Kitikiti

2025-09

University of the Witwatersrand, Johannesburg

https://hdl.handle.net/10539/49901

Tsheoga, Kamogelo Kitikiti. (2025). Synthesis and Optimization of a Water Network Scheme for a Multipurpose Multiproduct Textile Dyeing Facility: An Industrial Case Study. [Master's dissertation, University of the Witwatersrand, Johannesburg]. WIReDSpace. https://hdl.handle.net/10539/49901

Downloaded from a DSpace repository. https://dspace.org/



UNIVERSITY OF THE
WITWATERSRAND,
JOHANNESBURG

Synthesis and Optimization of a Water Network Scheme for a Multipurpose Multiproduct Textile Dyeing Facility: An Industrial Case Study

Kamogelo Tsheoga (1422752)

Master of Science in Engineering by Research:

A dissertation submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Master of Science in Engineering.

Supervisor: Professor Thokozani Majozi

05 September 2025

Declaration

I declare that this dissertation is my own unaided work. It is being submitted for the Degree of Master of Science in Chemical Engineering to the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination to any other university.



(Signature of Candidate)

5th day of September year 2025

Acknowledgements

"*You thrill me, Lord, with all you have done for me! I sing for joy because of what you have done. O Lord, what great works you do!*" (Psalm 92: 4-5, NLT)

First and foremost, I would like to express my deepest gratitude to my supervisor Professor Thokozani Majozi, for his invaluable mentorship, guidance, and unwavering support throughout this journey. I am especially grateful for the many opportunities he has afforded me throughout my research, which have greatly contributed to my academic and personal growth. His patience and expertise have been instrumental in shaping my work. I am also deeply grateful to Professor Mamdouh Gadalla, for his insightful guidance and taking the time to review my dissertation.

I extend my heartfelt appreciation to my colleagues from the Sustainable Process Engineering (SPE) research group and my dear friends, for their constructive feedback, constant motivation, and camaraderie. Their encouragement has made this journey both rewarding and enjoyable. I am also grateful to Dr. Natsayi Chiwaye, whose guidance and resourcefulness when assisting with one of the models contributed significantly to this dissertation. A sincere thanks to Professor Rajagopalan Srinivasan, for his mentorship and guidance during my semester exchange program at the Indian Institute of Technology (IIT) Madras, in Chennai, India. I am also grateful to my colleagues and acquaintances from IIT Madras, for their warm hospitality, and enriching experiences during my time there.

Most importantly, I am deeply thankful to my family – *Nape 'a Morwakoma le Pheladi* and *Pheladi 'a Phaahle le Hlapjadi*, and my amazing sisters (Itu, Khomo, Thato and Thabi) – for their unconditional love, sacrifices, and belief in me. Their support has been my foundation, and I could not have done this without them. A special mention to my partner, Phathutshedzo, whose endless encouragement, understanding, and unwavering faith in me has been my greatest source of strength and motivation. Your love and support have made this journey all the more meaningful.

I would like to acknowledge the National Research Fund (NRF) of South Africa for their financial support, which made this research possible.

Abstract

The textile industry plays a critical role in global economic and social development as it provides essential goods and employment to millions worldwide. However, it is also among the most environmentally taxing industries, characterized by intensive water consumption and significant wastewater emissions. These challenges not only contribute to severe environmental degradation but also pose significant risks to public health due to the toxic substances released into water bodies. Traditional end-of-pipe solutions, which treat wastewater after it is generated, offer limited relief and incur considerable financial and operational costs, making them unsustainable in the long term. The growing urgency to address these issues has highlighted the importance of process integration, which takes a holistic approach by optimizing resource use within the production system in order to reduce waste and improve efficiency at the source. Thus, process integration offers a transformative pathway to address water-related challenges of the textile industry by implementing strategies such as direct and indirect water reuse, recycle, and regeneration.

This work presents a Mixed Integer Non-linear Programming model for the optimization of both water usage and makespan in a multipurpose multiproduct textile dyeing process. The proposed approach integrates production scheduling with wastewater minimization within a simultaneous optimization framework with the aim of reducing freshwater consumption and wastewater generation while maintaining minimal impact on production schedules. Key factors such as product demand, water requirements, and wastewater generation are incorporated to design a sustainable water management strategy that balances operational, economic, and environmental constraints. The computational feasibility of the framework was validated using advanced solvers such as CONOPT3 and DICOPT. However, the integration of water reuse into production scheduling introduced significant complexity, particularly for larger scenarios.

An industrial case study was used to demonstrate the applicability of the proposed model. Results revealed significant water savings of up to 60.4% within the water network, with a manageable 77-minute deviation from the optimal makespan. These results underscore the effectiveness of a unified approach in identifying and utilizing water reuse and recycling opportunities while maintaining production schedules with minimal disruption. This highlights the model's potential for sustainable and efficient water management in industrial processes.

Table of Contents

Declaration ii
Acknowledgements iii
Abstract iv
List of Figures vii
List of Tables viii
Chapter 1 Introduction 1-1
1.1. Background 1-1
1.2. Motivation 1-3
1.3. Objectives 1-5
1.4. Problem Statement 1-6
1.5. Dissertation Structure 1-6
1.6. References 1-8
Chapter 2 Literature Review 2-1
2.1 Introduction 2-1
2.2 Batch Processes 2-2
2.2.1 Definition and Background 2-2
2.2.2 Storage Operational Philosophies 2-3
2.2.3 Recipe Representation 2-4
2.2.4 Time Representation 2-5
2.3 Batch Scheduling 2-9
2.3.1 Scheduling Techniques for Batch Processes 2-9
2.3.2 Scheduling of Multistage Batch Processes 2-15
2.3.3 Scheduling of Multistage Multiproduct Batch Processes 2-21
2.4 Water Network Synthesis (WNS) 2-24
2.4.1 Insight-Based Methods for Water Network Synthesis 2-26
2.4.2 Mathematical Optimization Techniques for Water Network Synthesis 2-29
2.4.3 Wastewater Minimization in Multipurpose Batch Processes 2-31
2.4.4 Wastewater Minimization in Textile Plants 2-33
2.5 Challenges in Water Network Synthesis 2-35
2.5.1 Non-linearity and Non-convexity 2-35
2.5.2 Binary Variables 2-35

2.5.3 Multi-Objective Optimization 2-36
2.6 Background on Mathematical Modelling 2-37
2.6.1 Classification of Optimization Problems 2-39
2.6.2 Convexity vs. Non-convexity 2-41
2.6.3 Linearization of Bilinear terms 2-42
2.6.4 Solution Methods and Solvers for MINLP Problems 2-45
2.7 Summary 2-51
2.8 References 2-51
Chapter 3 Model Formulation 3-1
3.1 Introduction 3-1
3.2 Superstructure Representation 3-1
3.3 Assumptions 3-3
3.4 Nomenclature 3-3
3.5 Mathematical Model 3-5
3.5.1 Production Schedule Model 3-6
3.5.2 Water Network Model 3-10
3.6 References 3-15
Chapter 4 Industrial Case Study 4-1
4.1 Introduction 4-1
4.2 Case Study 4-1
4.3 Results and Discussion 4-3
4.4 References 4-15
Chapter 5 Limitations 5-1
5.1 Introduction 5-1
5.2 Limitations 5-1
5.2.1 Computational Intensity 5-1
5.2.2 Further Water Saving Potentials 5-3
5.3 References 5-3
Chapter 6 Conclusion and Future Work 6-1

List of Figures

Figure 1.1. Industrial freshwater withdrawal as a percentage of the total available freshwater resources per country (UN-Water, 2024)..... 1-1
Figure 1.2. Schematic layout showing water motion through a textile plant 1-5
Figure 1.3. Dissertation layout..... 1-7
Figure 2.1. Batch process configurations..... 2-2
Figure 2.2. State Task Network representation..... 2-4
Figure 2.3. Resource Task Network representation..... 2-5
Figure 2.4. State Sequence Network representation 2-5
Figure 2.5. Division of time in discrete time representation..... 2-6
Figure 2.6. Slot-based representation..... 2-7
Figure 2.7. Global event-based representation 2-8
Figure 2.8. Unit-specific event-based representation 2-8
Figure 2.9. A general depiction of single-stage and multi-stage processes 2-17
Figure 2.10. Direct reuse/recycle of water from operation A to B 2-25
Figure 2.11. Indirect reuse/recycle of water from operation A to B..... 2-25
Figure 2.12. Classification of systematic synthesis methods..... 2-30
Figure 2.13. An illustration of the numerical optimization framework (Diwekar, 2020) ... 2-38
Figure 2.14. Linear and nonlinear programming problems (Diwekar, 2020)..... 2-40
Figure 2.15. Illustration of convex and non-convex sets (Williams, 2013; Diwekar, 2020)2-41
Figure 2.16. Local optima and global optimal solution in non-convex NLP problems..... 2-42
Figure 3.1. General superstructure representation of the water network..... 3-2
Figure 3.2. Components of the proposed formulation 3-6
Figure 3.3. Water Network around unit j, shade i..... 3-10
Figure 3.4. Water network around the storage tanks 3-13
Figure 4.1. Production recipe..... 4-1
Figure 4.2. Optimal Production Schedule before water integration 4-4
Figure 4.3. Gantt chart for Scenario A without storage tanks 4-6
Figure 4.4. Gantt chart for Scenario A with storage tanks..... 4-7
Figure 4.5. Gantt chart for Scenario B without storage tanks..... 4-9
Figure 4.6. Gantt chart for Scenario B with storage tank 4-10

List of Tables

Table 2.1. Classes of optimization problems (Dutta, 2016) 2-39

Table 2.2. MINLP Solvers (Kronqvist et al., 2019)..... 2-49

Table 4.1. Production recipe for each stage..... 4-2

Table 4.2. Model statistics 4-5

Table 4.3. Comparative results for Case Study..... 4-11

Table 4.4. Comparative analysis of multipurpose multiproduct batch scheduling models . 4-13

Chapter 1

Introduction

1.1. Background

The problem of water scarcity is a long-standing global conundrum, and its severity is increasing. According to the United Nations, by 2030, the global water demand is expected to exceed supply by 50% which has far-reaching implications for various sectors, including industries, communities, and natural ecosystems (World Economic Forum, 2023). Industries, in particular, face significant challenges in minimizing freshwater usage and wastewater generation. The 2021 United Nations World Water Development Report reveals that industry accounts for 19% of global water consumption, and a staggering 80% of global industrial wastewater is discharged untreated, polluting our water sources and harming ecosystems (UN-Water, 2021). Figure 1.1 further emphasizes this point, depicting the varying degrees of water stress experienced by different countries, with some countries exhibiting high levels of industrial water use relative to their available freshwater resources.

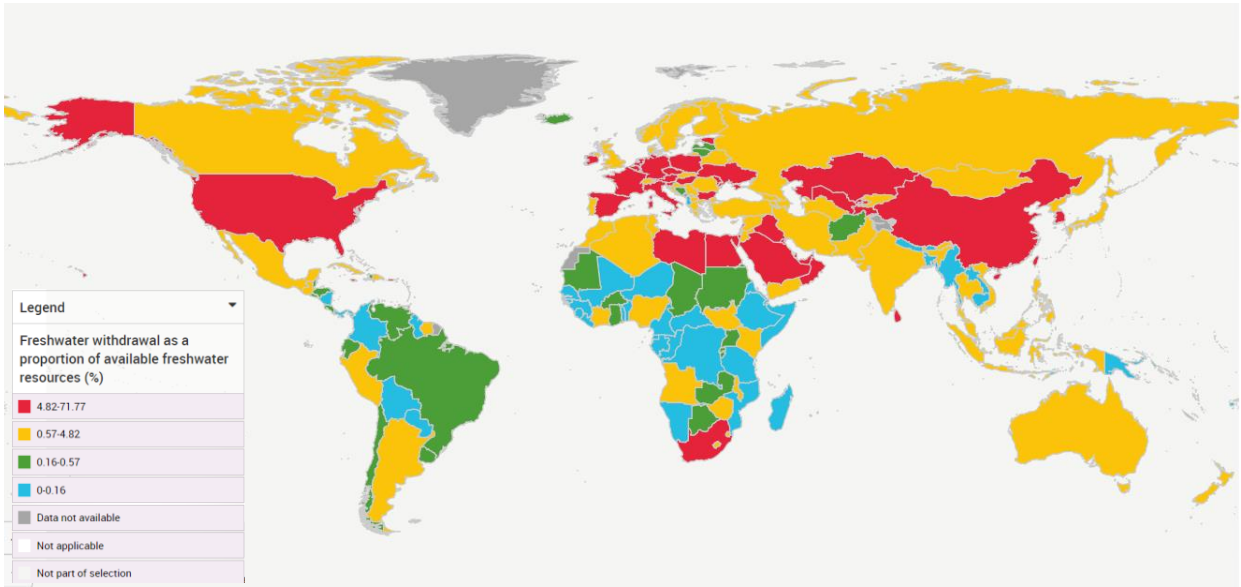


Figure 1.1. Industrial freshwater withdrawal as a percentage of the total available freshwater resources per country (UN-Water, 2024)

The water usage challenge is particularly acute in the textile sector, where diverse production processes are employed to manufacture different types of textile products. Textile plants are typically multistage and multiproduct in nature, i.e., different products (each with distinct recipes) are produced in multiple units configured in parallel. These plants often face complexities in wastewater treatment due to the varied composition of wastewater streams, fluctuating production demands, and limited space for treatment infrastructure. This has made the textile industry to be notorious for its high-water usage, from fibre cultivation to dyeing and finishing processes throughout its various stages of production. It ranks as the second-largest water consumer globally, following agriculture (Turk, 2023). According to the World Bank, the textile industry alone consumes approximately 79 billion cubic meters of water annually (European Parliament, 2024). Furthermore, it accounts for nearly 20% of global industrial water pollution due to the discharge of untreated wastewater containing harmful chemicals and dyes (Bailey et al., 2022).

In developing countries, where water scarcity has reached a critical level, the effects of industrial water usage, particularly in the textile sector, are notably severe. For example, in water-stressed countries such as South Africa, 98.6% of the total available water resources are exhausted (GreenCape, 2023). According to recent data from the South African Department of Water and Sanitation, about 27% of the available water supply is allocated to industries, with a significant portion used in manufacturing processes, including textile production (Department of Water and Sanitation, 2020). Although the textile industry contributed around 2% to the country's total manufacturing Gross Domestic Product (GDP) in 2022, its substantial water consumption and pollution add additional pressure to already strained water resources (PwC, 2023; Trade & Industry Policy Strategies, 2024). This, along with the increased global awareness of sustainable development, highlights the urgent need to reduce the water footprint of such industries by adopting sustainable water management practices to mitigate the impacts of industrial water usage on freshwater availability and quality. Additionally, stringent regulatory requirements and environmental standards mandate industries to minimize their environmental footprint and ensure compliance with wastewater discharge regulations.

In response to the challenges posed by wastewater generation in batch processes, researchers have increasingly sought to develop strategies for its minimization. A prevalent approach within the existing literature is process integration. This holistic approach emphasizes the

interconnectedness of a process to maximize the efficient use of process resources, such as equipment, energy, water and other utilities, with the ultimate goal of optimizing the overall value (El-Halwagi, 1997). To address water minimization challenges, process integration focuses on identifying opportunities for direct or indirect water reuse and recycling within the production system by matching wastewater with specific contamination levels to meet the input requirements of subsequent processes (Adekola & Majozi, 2017). In direct reuse, an effluent stream from a processing unit is used in another processing unit. Direct recycle refers to the re-entering of the wastewater stream to the same process unit in which it is produced (Gouws et al., 2010). Conversely, indirect reuse/ recycle permits the intermediate storage of wastewater prior to its subsequent utilization (Adekola & Majozi, 2011).

Research on wastewater minimization has predominantly targeted continuous water-using operations, with less emphasis on batch processes, which must consider both time constraints and contaminant concentration limits. Despite a growing interest in optimizing batch facilities, there remains a notable gap in applying process integration techniques specifically for wastewater minimization in multipurpose, multiproduct textile plants. Existing literature largely focuses on treatment methods to reduce effluent toxicity in textile wastewater (Ciardelli & Ranieri, 2001; Cinperi et al., 2019; Kant, 2012; Punzi et al., 2015), with limited studies addressing strategies for optimizing water usage within this water-intensive industry. This dissertation directly addresses this gap by developing a mathematical model designed to synthesize and optimize a water network scheme tailored to the complex requirements of textile plants.

1.2. Motivation

Water scarcity remains an unignorable problem worldwide, especially in water-stressed countries like South Africa, India, England, Iran and Mexico (Kuzma et al., 2023). The textile industry, being the second-largest consumer of freshwater worldwide after agriculture and a significant polluter second only to the oil industry, plays a significant role in this crisis (Smarter Business Editorial Team, 2020). Textile production involves multiple stages, each consuming large amounts of water and releasing various toxic pollutants, from fibre preparation to fabric dyeing and garment finishing (Vineta et al., 2014). While batch processing in textile plants provides flexibility, it also creates challenges for wastewater management, generating large

volumes of water contaminated with chemicals and dyes that are difficult to treat. Traditionally, research on batch processes has focused on conventional batch operations or end-of-pipe treatment methods to manage wastewater after it is generated. However, few studies have explored process integration strategies for multipurpose, multiproduct textile facilities, which face unique challenges due to their diverse processes and high levels of water and chemical consumption. With the rising demand for textiles driven by the fashion industry, both the volume of textile wastewater and its environmental impact are increasing significantly (Khan & Malik, 2014). This growing pollution not only poses a major environmental concern but also directly threatens water availability and public health. Therefore, proactive strategies that reduce wastewater production at the source, rather than relying solely on treatment post-generation, have become essential.

A unique characteristic of the textile dyeing process is that wastewater tends to become progressively less contaminated as it moves through each stage thus creating valuable opportunities for water reuse and recycling within the process. Specifically, effluent from each stage can potentially be redirected to the same or even previous stages, provided that the contaminant levels in the effluent meet the required inlet concentration of the receiving stage. As shown in Figure 1.2, the typical configuration involves producing darker fabrics further down the line. This parallel layout enables effluents from one stage in a given line to be reused in earlier stages of other lines, provided the effluent concentration aligns with the inlet requirements of the subsequent stages. By capitalizing on this arrangement, the industry can reduce freshwater usage and waste generation.

Environmental sustainability is both a moral duty and a regulatory requirement, with strict guidelines governing wastewater discharge. Additionally, optimizing water use has recognized economic benefits, including cost savings and improved productivity. Scheduling textile operations in these plants is particularly demanding, as each stage has its own water requirements and distinct chemical requirements which produce varying effluent profiles. These operational complexities mean that efficient wastewater management must account for the timing, sequence, and coordination of each process to optimize water reuse and reduce wastewater. This research aims to address these complexities by developing a robust model for wastewater minimization in multiproduct multipurpose textile plants. The proposed mathematical model for the water-intensive textile industry integrates two sub-problems:

makespan minimization (Production Scheduling Model) and wastewater minimization (Water Network Model).

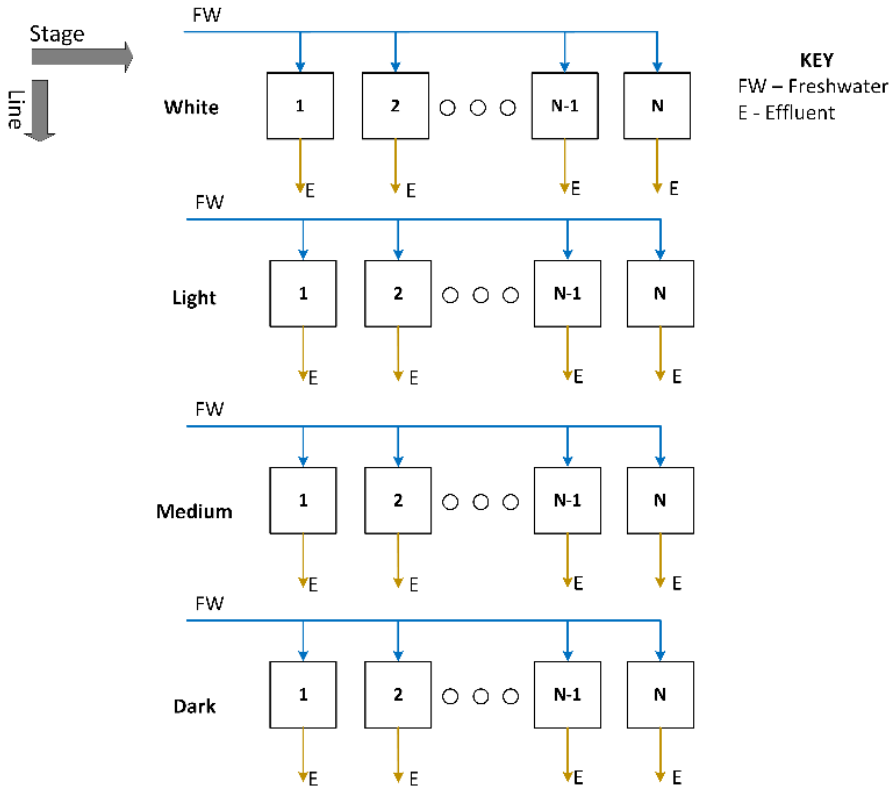


Figure 1.2. Schematic layout showing water motion through a textile plant

1.3. Objectives

The objectives of the study are stated as follows:

- To develop a scheduling model for a multipurpose multiproduct textile facility, which is responsible for identifying the production sequence for the given demand of products within an optimized makespan.
- To develop a water model responsible for identifying inter-stage and inter-unit water-saving opportunities in the production sequence.
- To integrate the models mentioned above into one formulation that allows for the simultaneous optimization of makespan and water usage.

Using an industrial case study sourced from a textile plant in India, this work will demonstrate the applicability of the proposed model as well as validate it with the realized water savings.

Although the emphasis is on showing results for the textile plants, the methods are valid for multistage multiproduct batch plants with a similar parallel configuration.

1.4. Problem Statement

The multipurpose multiproduct batch problem that will be addressed can be stated as follows. Given:

- Production scheduling data, including recipes for each product, the available processing units, and their capacities,
- The task processing time and production demand
- The threshold concentrations of each contaminant,
- Water network data, including contamination load in each unit, water requirements in each unit, design capacity limits, etc.

Determine the optimal schedule that achieves minimum wastewater production in the least time.

1.5. Dissertation Structure

This dissertation consists of six chapters. Figure 1.3 depicts the layout of the dissertation. Chapter 1 provides an overview of the study by describing the background, motivation, problem statement and the scope of this research work. The subsequent sections of the dissertation are structured as follows. Chapter 2 provides a literature-based background for the research and examines previous studies on wastewater minimization in batch processes through various techniques, with a specific emphasis on mathematical modelling. The developed model is detailed in Chapter 3 and validated in Chapter 4 using results obtained when the model was applied to an industrial case study from India. Chapter 5 gives the limitations of the work and provides recommendations for future research in this area. The conclusions drawn are presented in Chapter 6.

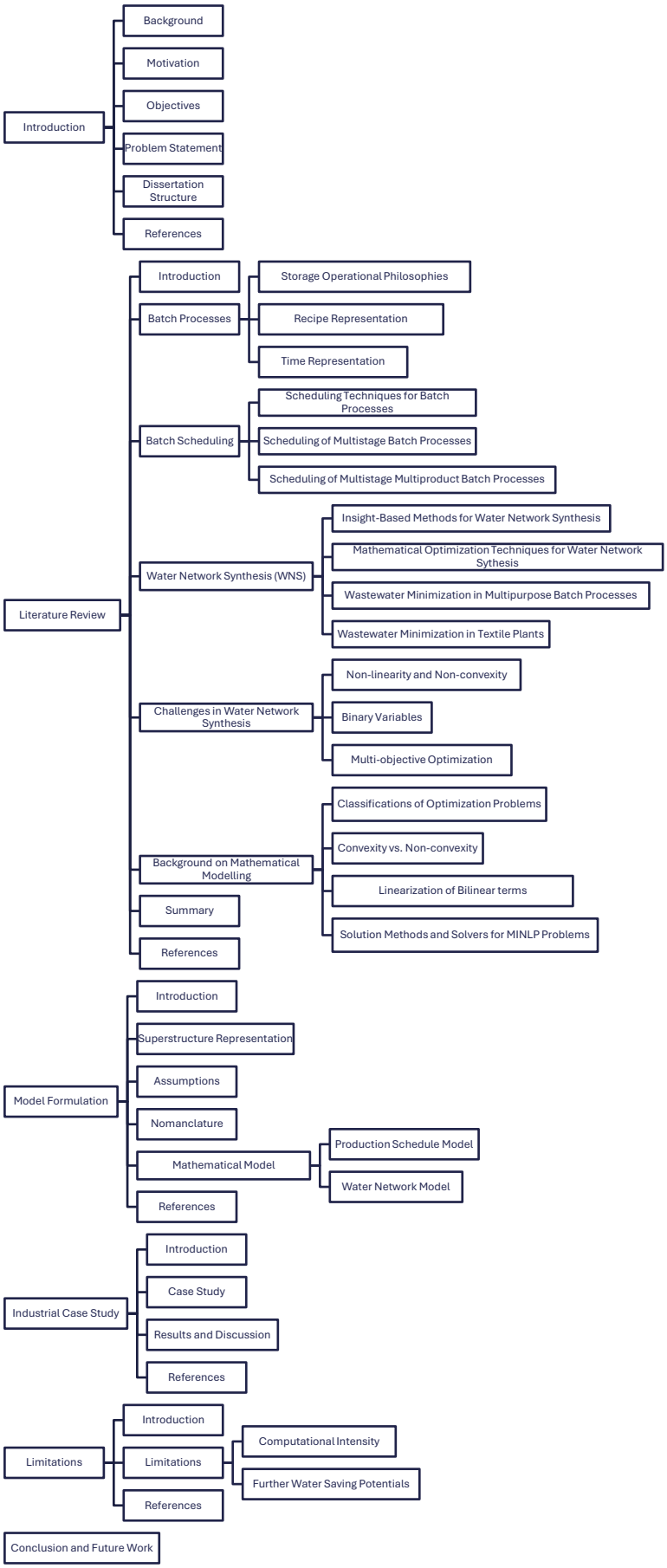


Figure 1.3. Dissertation layout

1.6. References

Adekola, O., & Majozi, T. (2011). Wastewater minimization in multipurpose batch plants with a regeneration unit: Multiple contaminants. *Computers and Chemical Engineering*, *35*(12), 2824–2836. https://doi.org/10.1016/j.compchemeng.2011.04.008

Adekola, O., & Majozi, T. (2017). Wastewater minimization in batch plants with sequence dependent changeover. *Computers and Chemical Engineering*, *97*, 85–103. https://doi.org/10.1016/j.compchemeng.2016.11.016

Ciardelli, G., & Ranieri, N. (2001). The treatment and reuse of wastewater in the textile industry by means of ozonation and electroflocculation. *Water Research*, *35*(2), 567–572. https://doi.org/10.1016/S0043-1354(00)00286-4

Cinperi, N. C., Ozturk, E., Yigit, N. O., & Kitis, M. (2019). Treatment of woolen textile wastewater using membrane bioreactor, nanofiltration and reverse osmosis for reuse in production processes. *Journal of Cleaner Production*, *223*, 837–848. https://doi.org/10.1016/j.jclepro.2019.03.166

Department of Water and Sanitation. (2020). *Strategic Plan (Vote 41)*. www.dws.gov.za

El-Halwagi, M. M. (1997). *Pollution Prevention through Process Integration: Systematic Design Tools*. Academic Press.

European Parliament. (2024a). *The impact of textile production and waste on the environment (infographics)*. https://www.europarl.europa.eu/topics/en/article/20201208STO93327/the-impact-of-textile-production-and-waste-on-the-environment-infographics

European Parliament. (2024b, March 21). *The impact of textile production and waste on the environment (infographics)*.

Global Fashion Agenda and The Boston Consulting Group. (2017). *Pulse of the Fashion Industry Report*.

Gouws, J. F., Majozi, T., Foo, D. C. Y., Chen, C. L., & Lee, J. Y. (2010). Water minimization techniques for batch processes. *Industrial and Engineering Chemistry Research*, *49*(19), 8877–8893. https://doi.org/10.1021/ie100130a

GreenCape. (2023). *2023 WATER MARKET INTELLIGENCE REPORT*. www.greencape.co.za.

Kant, R. (2012). Textile dyeing industry an environmental hazard. *Natural Science*, *04*(01), 22–26. https://doi.org/10.4236/ns.2012.41004

Khan, S., & Malik, A. (2014). Environmental and Health Effects of Textile Industry Wastewater. In A. Malik, E. Grohmann, & R. Akhtar (Eds.), *Environmental Deterioration and Human Health: Natural and anthropogenic determinants* (pp. 55–71). Springer Netherlands. https://doi.org/10.1007/978-94-007-7890-0_4

Kuzma, S., Saccoccia, L., & Chertock, M. (2023, August 16). *25 Countries, Housing One-Quarter of the Population, Face Extremely High Water Stress*. World Resources Institute.

Punzi, M., Nilsson, F., Anbalagan, A., Svensson, B. M., Jönsson, K., Mattiasson, B., & Jonstrup, M. (2015). Combined anaerobic-ozonation process for treatment of textile wastewater: Removal of acute toxicity and mutagenicity. *Journal of Hazardous Materials*, *292*, 52–60. https://doi.org/10.1016/j.jhazmat.2015.03.018

PwC. (2023, July 17). *SA's manufacturing sector must increase 4IR tech adoption, or risk falling behind global competitors.* PwC. https://www.pwc.co.za/en/press-room/manufacturing-workforce.html#:~:text=In%202022%2C%20the%20sector%20contributed,changing%20the%20face%20of%20manufacturing.

Smarter Business Editorial Team. (2020, January 13). *Top 5 Industries with the Highest Water Consumption*.

Trade & Industry Policy Strategies. (2024). *Industry Study: Clothing and Textiles*.

Turk, V. (2023, October 2). *Fast Fashion: Polluting our Water Millions of Clothing Items at a Time*. GW ENVIRONMENTAL AND ENERGY LAW.

UN-Water. (2021). *The United Nations World Water Development Report 2021: Valuing Water*. United Nations Educational, Scientific and Cultural Organization.

Vineta, S., Silvana, Z., Sanja, R., & Golomeova, S. (2014). Methods for Waste Waters Treatment in Textile Industry. *International Scientific Conference "Unitech Gabrovo," November*, 248–252.

Chapter 2

Literature Review

2.1 Introduction

In recent decades, the optimization of batch processes has become a crucial focus in chemical engineering, driven by the need for greater efficiency, resource conservation, and sustainability. In particular, batch process optimization has become relevant for industries where resource-intensive operations, such as water usage and wastewater treatment, play a substantial role. This section provides a detailed overview of batch scheduling and process integration, with a focus on wastewater minimization through mathematical modelling. The literature review is organized into four key areas, beginning with a general overview of batch processes and moving to a discussion of specialized optimization techniques for wastewater minimization.

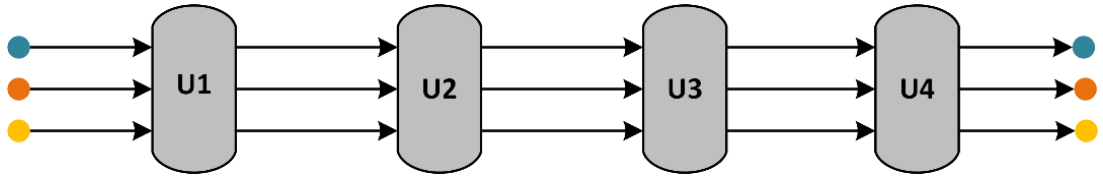
The first section (section 2.2) gives a clear overview of batch processes, highlighting their importance and applications in industry. Section 2.3 explores batch scheduling, reviewing existing literature and methodologies that address the complexities of coordinating multiple stages and tasks in batch production. Building on these scheduling concepts, Section 2.4 focuses on water network synthesis and explores approaches to wastewater minimization, specifically within both conventional batch facilities and textile plants, where water usage is intensive. The chapter concludes with an overview of mathematical optimization, presenting relevant techniques and algorithms for solving Mixed-Integer Nonlinear Programming (MINLP) problems, which are essential for optimizing complex batch processes.

2.2 Batch Processes

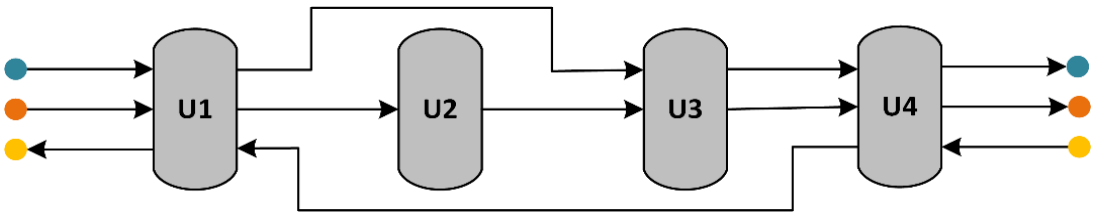
2.2.1 Definition and Background

A batch process is a time-dependent process which involves a series of discrete operations executed in defined stages to produce a specified quantity of product. Batch processes involve the manufacture of a wide variety of diverse specialized products in relatively small quantities. In many instances, manufacturing necessitates multiple sequential processing phases and encompasses diverse intermediary products (Grunow & Lehmann, 2002). Batch processes are commonly employed in industries such as pharmaceuticals, chemicals, and textiles, where products are manufactured in discrete batches rather than in continuous streams.

In batch production, multiproduct and multipurpose configurations are the two most prevalent types. Multiproduct plants are characterised by different products that are processed in the same sequence of unit operations, following the same path from raw materials to final products (Majozi et al., 2017). Production occurs through a succession of production campaigns for each product, and only one product is manufactured at a time (Modi & Karimi, 1989). Such facilities are best suited for products with identical and fixed recipes. Figure 2.1(a) depicts a typical configuration of multiproduct batch plants. Textile production, as an example of a multiproduct batch plant, involves a complex process with many stages in the production sequence.



(a) Multiproduct batch



(b) Multipurpose batch

Figure 2.1. Batch process configurations

A certain production demand would need to be met in the textile facility, rendering the process multistage and multiproduct. In multipurpose batch processes, various batches of products are produced through distinct production routes, as depicted in Figure 2.1 (b). Multipurpose batch processes provide increased versatility in terms of product variety, production volume, and the ability to process a range of recipes interchangeably within a single piece of equipment (Grunow & Lehmann, 2002).

2.2.2 Storage Operational Philosophies

Storage operating philosophies are categorized according to the level of availability and the capacity to store intermediates, by-products, and/or products. According to the no intermediate storage (NIS) operational philosophy, the production line is the sole storage option, and intermediates can only remain in the same unit where they are generated (Thornton & Hunsucker, 2004). This storage policy is particularly important when operational space is limited in a batch production facility. The finite intermediate storage (FIS) operational philosophy involves storing intermediate goods in a tank that has a limited capacity (Kim et al., 1996). Unlimited intermediate storage (UIS) is the limitless FIS where the availability of storage is always guaranteed. Occasionally, different operations in the facility may utilize the same storage tank(s), and there are instances when idle processing units might serve as storage for intermediates and/or products (Jung et al., 1996). These are often known as common intermediate storage (CIS) and process intermediate storage (PIS), respectively. Such tanks must be cleaned in order to preserve the integrity of the products. Batch plants provide intrinsic flexibility, and thus, the majority of them utilize a combination of two or more of the storage operating philosophies outlined above (Ku & Karimi, 1990). The term used to describe this operational philosophy is mixed intermediate storage (MIS).

Some operational philosophies are classified based on the stability of the intermediate product(s), in addition to the criteria mentioned above. The zero wait (ZW) approach is ideal for handling unstable intermediates that must be immediately utilized in the subsequent job without the requirement for any intermediate storage. Finite wait (FW) and unlimited wait (UW) operational philosophies are suitable for managing intermediates that can wait for a limited duration and intermediates that can wait for longer periods, respectively (Majozi, 2010).

2.2.3 Recipe Representation

A recipe representation illustrates the sequence of tasks necessary to transform raw materials into a desired product. Amongst other things, it is necessary for the development of efficient mathematical models. The most used representations developed are the State Task Network (STN), Resource Task Network (RTN), State Sequence Network (SSN) and the Schedule Graph (S-Graph).

Kondili et al. (1993) introduced the concept of the STN to address the challenges that emerge in real-world scenarios, such as the splitting, mixing, and recycling of material. The representation employs two distinct nodes: state nodes, which represent raw materials, intermediate materials, or final products, and the task nodes, which represent a unit operation/ processing equipment. Figure 2.2 shows the STN representation of an illustrative example. The states in the example consist of two feeds, one intermediate product, and one final product. Additionally, the two tasks represent the two steps that the processing facility uses to manufacture the final product.

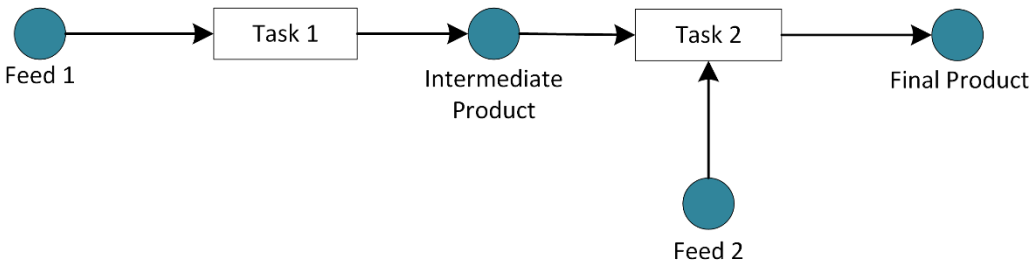


Figure 2.2. State Task Network representation

Although the STN representation is beneficial for facility representation, it only considers raw materials and products which restricts its applicability to real-life problems where other resources should be considered. Pantelides (1994) thus enhanced the STN representation by introducing RTN. The RTN expands the state nodes by simultaneously representing other resources, such as storage, human resources, transportation facilities, material transfer, and utilities. The example depicted in Figure 2.3 contains two resources which are necessary for both tasks.

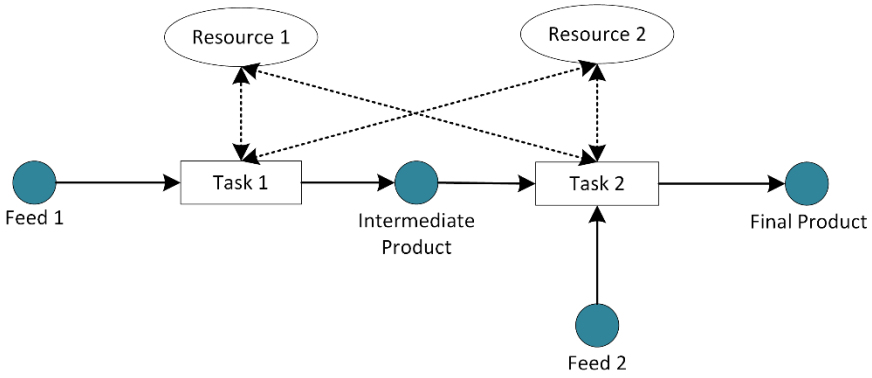


Figure 2.3. Resource Task Network representation

In 1998, Sanmartí et al. introduced the S-graph, which takes into consideration the specific characteristics of chemical processes in scheduling. The nodes in the graphs represent production tasks, and the arcs represent the hierarchical link between them. However, this representation is restricted to scheduling problems with NIS and UIS transfer policies and does not incorporate any heuristics to guide the branching procedure (Sanmartí et al., 1998). Majozi and Zhu (2001) introduced the SSN, a strategy aimed at reducing binary variables without utilising a variable reduction technique. The SSN representation simply includes state nodes, and tasks that take place within a unit are inferred from the changes in states (Majozi, 2010).

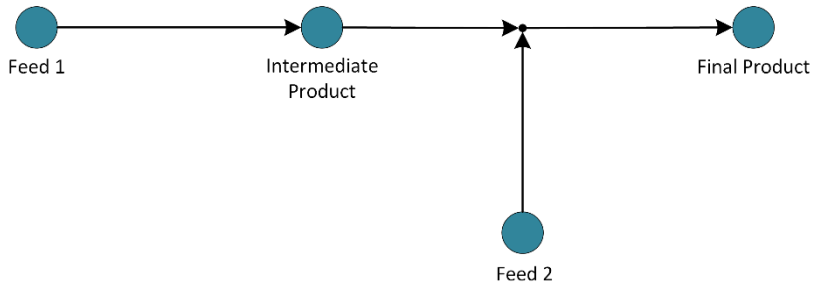


Figure 2.4. State Sequence Network representation

2.2.4 Time Representation

The type of time representation used in a model significantly affects computational efficiency as well as the accuracy of results. Different time representations can result in models of varying complexity and flexibility. Generally, time formulations can be categorized into two approaches: discrete-time and continuous-time.

2.2.4.1 Discrete-Time Representations

In this approach, the time horizon is divided into a number of fixed time intervals of known length before optimization takes place. Its main advantage is that it provides a time reference grid for all operations which compete for shared resources such as equipment (Janak et al., 2004). Méndez et al. (2006) added that the discrete-time formulation is efficient, adaptable, and convenient for a wide variety of industrial applications due to its reduced complexity.

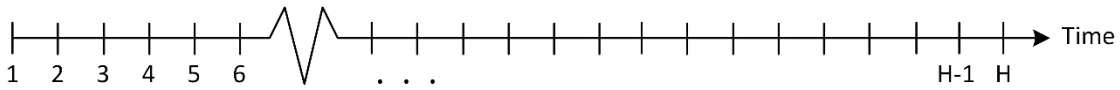


Figure 2.5. Division of time in discrete time representation

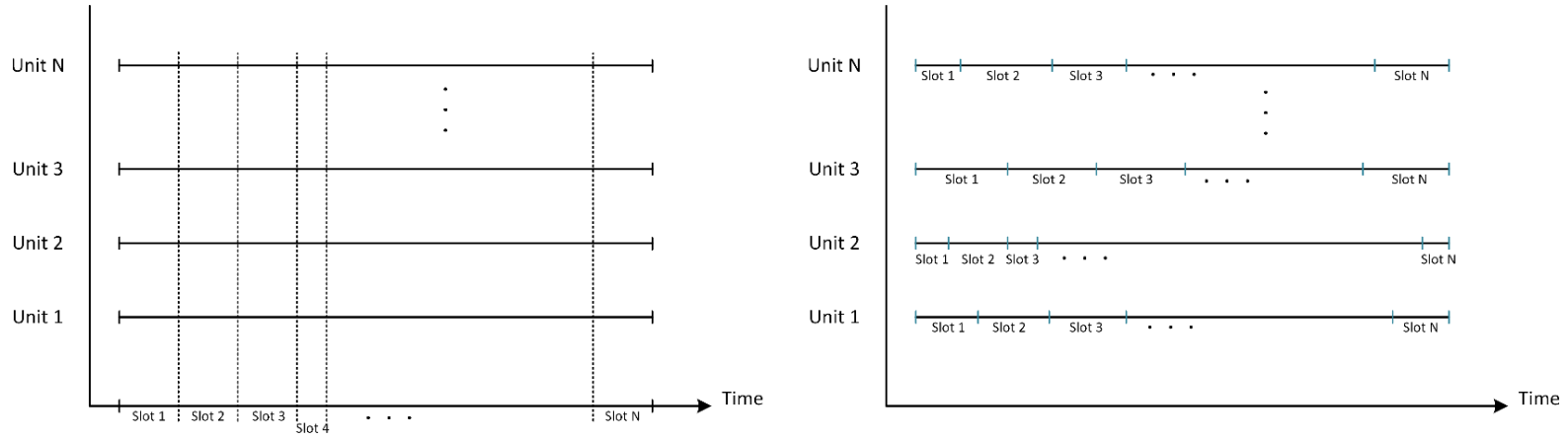
The main drawbacks to discrete time representations are their inability to accurately reflect reality and their tendency to overlook the true optima as a result of constraining time to be a fixed parameter. The selection of the duration with uniform time intervals presents a trade-off between the accuracy of the solution and the required computational efforts caused by a large number of binary variables (Floudas & Lin, 2004). Consequently, in order to precisely depict time, it may be necessary to use smaller time intervals with more binary variables (Majozi, 2010).

2.2.4.2 Continuous-Time Representations

Numerous continuous-time formulations have been developed to address the limitations of discrete time representation. These formulations offer significant potential for the development of more accurate and computationally efficient models (Floudas & Lin, 2004). Unlike discrete time representation, continuous time methods distribute the time horizon of interest into flexible intervals. This allows for precise determination of task start and end times, thereby enhancing model accuracy (Schilling & Pantelides, 1996). Based on how the time distribution is set up, continuous-time representations are broadly classified into three distinct categories: slot-based, global event-based, and unit-specific event-based representations.

A. Slot-Based Representation

In the slot-based representation, time is divided into ordered blocks of unknown variable lengths, where tasks are scheduled within these defined time slots. The optimization problem then determines the lengths of each time. This representation can be classified into synchronous and asynchronous representations. The synchronous representation shown in Figure 2.6 (a) defines common slots across all units such that shared resources in network batch processes are easier to handle. On the other hand, asynchronous representation is more flexible as it allows the postulated slots to differ from one unit to another. The idea of using slots in a continuous-time representation helps in developing efficient scheduling models without decoupling tasks from units, which results in fewer binary variables.



(a) Synchronous representation (b) Asynchronous representation

Figure 2.6. Slot-based representation

B. Global Event-Based Representation

This representation uses a set of event points that are common across all units and are defined for either the beginning or end (or both) of each task in each unit. In contrast to the discrete time formulation, the exact position of each event point is not predetermined.

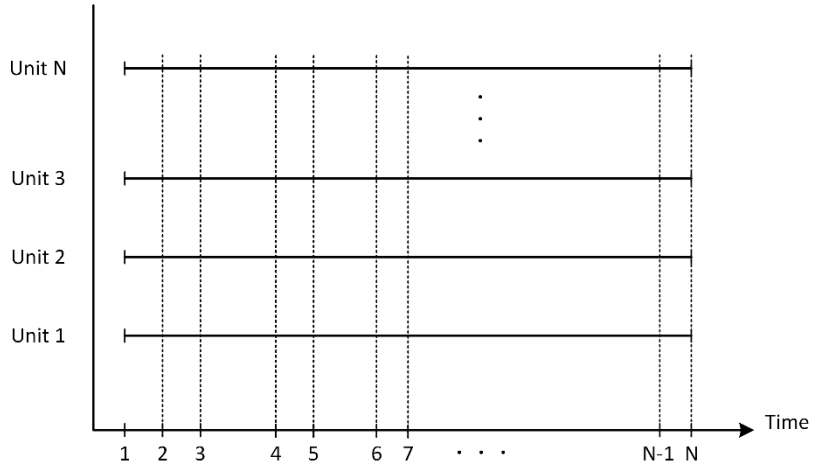


Figure 2.7. Global event-based representation

C. Unit-Specific Event-Based Representation

Similar to the global event-based representation, the unit-specific event-based representation divides the scheduling horizon using different event points (Ierapetritou & Floudas, 1998). Each shared unit has its own set of event points, which enables different tasks to begin at different times within the same event point.

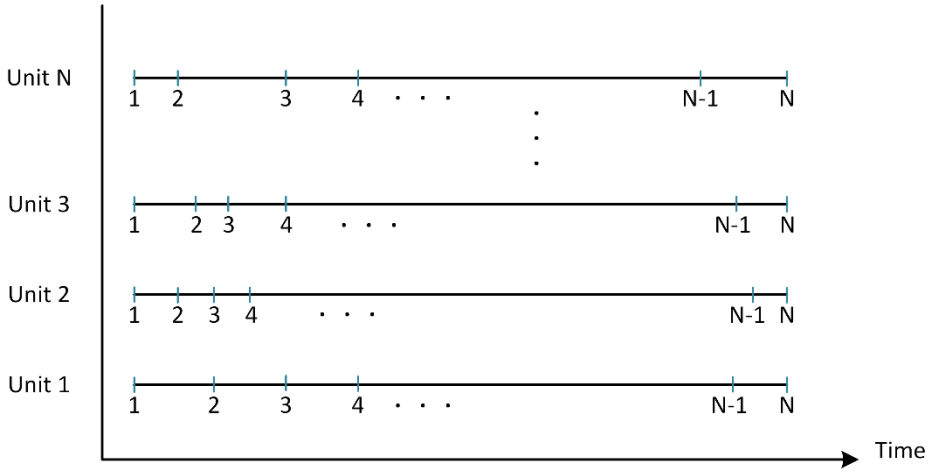


Figure 2.8. Unit-specific event-based representation

For each unit, the scheduling horizon is split into several event points. In other words, the processing tasks in unit 1 are assigned different event points, suggesting that the activities cannot occur simultaneously. Event points in unit 2 are not influenced by those in unit 1 and can therefore be represented by any notation, such as t_1, t_2, etc. Likewise, the same applies to

the remaining units. The heterogeneous location of event points results in fewer time points compared to the global event points (Méndez et al., 2006).

2.3 Batch Scheduling

Batch processes require the allocation of recipe steps to various available equipment units within a specific time frame due to the inherently discontinuous nature of resource utilization in batch plants. Hence, it is essential to monitor the instances along the time horizon when a task starts or ends, as well as when a resource is utilized. Batch scheduling is the allocation of resources, such as equipment, to various tasks and processes within a specific time interval to optimize production efficiency, minimize downtime, and ensure timely completion of all operations. Consequently, having precise time representation is crucial for effective batch scheduling.

2.3.1 Scheduling Techniques for Batch Processes

Scheduling constitutes a critical component in both the initial planning of plant development and in optimising the production performance of existing facilities (Méndez et al., 2006). It plays a pivotal role in determining the sequence and time horizon of batch processes, with a focus on resource conservation (Grunow & Lehmann, 2002). Through effective resource management and minimizing downtime, scheduling techniques play a crucial role in boosting overall production efficiency and lowering costs. The literature on batch processes presents a diverse array of scheduling techniques. These include insight-based techniques, stochastic, heuristic and metaheuristic approaches, constraint-guided and simulation-based strategies, as well as mathematical modelling techniques incorporating both conventional and advanced solution algorithms (Floudas & Lin, 2004). Over the past decades, insight-based techniques and mathematical modelling techniques have been the predominant approaches, with mathematical modelling being more frequent. Although heuristic and metaheuristic methods have gained prominence in recent years, insight-based techniques and mathematical modelling techniques continue to be fundamental because of their versatility, efficiency, ability to provide physical insights, and ability to provide valuable insights into complex batch scheduling problems.

Insight-based techniques help engineers to visually identify optimal usage points and efficiently design the resulting integrated network. Pinch analysis is a widely recognized graphical technique for identifying and minimizing resource consumption, particularly water and energy, in process systems. It offers valuable insights into optimal resource usage and is especially beneficial in industries focused on conservation and efficiency. However, pinch analysis has significant limitations when applied to batch scheduling. One major drawback is its inability to manage the temporal variability inherent in batch processes, where water and energy demands fluctuate due to the sequential and discrete nature of production stages (Foo, 2009). Unlike mathematical optimization techniques, pinch analysis lacks the capability to evaluate trade-offs between competing objectives rigorously.

The mathematical modelling approach effectively overcomes the limitations of the pinch analysis method and can resolve complex, large-scale industrial problems that are nonlinear and nonconvex (Budak Duhbacı et al., 2021). These techniques involve building models from mathematical equations that describe the system of interest and consider various constraints governed by the design, economics, and operational aspects of the process. Programming models, a subset of mathematical models, are distinguished by their focus on optimization. Unlike insight-based techniques which are usually limited to two dimensions, mathematical modelling techniques are able to handle complex problems due to their lack of dimensional limitations (Bazolana & Majozi, 2017). Furthermore, they are capable of managing scenarios with many uncertainties (Karuppiah & Grossmann, 2008).

Ghazouani et al. (2015) proposed a mathematical model for the heat integrated network problem, which provided more accurate results compared to the graphical approach used by Wan Alwi et al. (2011), thus highlighting the advantages of mathematical models over graphical techniques. Additionally, the model by Ghazouani et al. (2015) guarantees the identification of the global optimum solution, whereas the graphical method might overlook some optimal configurations. However, despite their flexibility in handling various dimensions, mathematical modelling approaches can face computational challenges, such as high CPU time, when solving complex optimization problems.

Mathematical programming is widely used to determine optimal production schedules within the processing industry (Grunow & Lehmann, 2002). This method is frequently chosen for

batch scheduling because it can effectively manage time as a variable and easily handle more complex objective functions (Faria & Bagajewicz, 2010). The nature of the problem typically dictates the specific mathematical programming method adopted. These methods include linear programming (LP), integer programming (IP), mixed integer linear programming (MILP), MINLP (discussed comprehensively in section 2.6) and hybrid methods that combine heuristics and exact algorithms.

The batch scheduling problem is most predominantly addressed using MILP due to its ability to guarantee a global optimum solution. The following review highlights significant advancements in optimizing batch scheduling through various formulations and algorithms, with a particular focus on MILP.

Kondili et al. (1993) introduced a method for scheduling batch processes using an STN representation and MILP formulation. The STN provides an unambiguous representation of complex processing networks, allowing for merging and splitting of batches. The MILP model optimizes equipment allocation, batch sizes, and resource utilization while considering various operational constraints. This formulation can accommodate different storage policies, utility limitations, and equipment unavailability. However, the main limitation is the large size of the resulting MILP problem due to the discrete-time representation. Kondili et al. (1993) acknowledged that even small examples require substantial computation time, with costs increasing rapidly for larger problems, which makes it computationally intensive to solve realistic scheduling problems.

Shah et al. (1993) tackled these computational issues by proposing four complementary measures to accelerate the solution process:

1. *Reformulating allocation constraints* to reduce the integrality gap between the LP relaxation and the MILP solution, thereby reducing the number of branch-and-bound iterations.
2. *Using reduced LP relaxations* to help in finding integer feasible solutions earlier in the search process.
3. *Implementing solution analysis techniques* to solve smaller LPs at each node of the branch-and-bound tree significantly speeds up the solution process.

4. *Employing different margins of optimality* to allow for trade-offs between solution quality and computational time.

The reformulated constraints resulted in a more tightened LP relaxation problem, and the solution techniques employed reduced the size of the relaxed LP, thus decreasing computational complexity. However, the use of larger margins of optimality to speed up solutions means potentially sacrificing some degree of optimality. Shah et al. (1993) also showed the applicability of the approach to more complex case studies, illustrating its effectiveness as a systematic scheduling tool for industrial-scale problems. Nonetheless, the discretized nature of the models proposed by Kondili et al. (1993) and Shah et al. (1993) inherently resulted in suboptimal outcomes.

Schilling & Pantelides (1996) presented a continuous-time scheduling formulation based on RTN and a branch-and-bound algorithm to mitigate the large integrality gap in continuous time formulations. Their approach divides the time horizon into variable-duration slots and introduces a branching strategy on both continuous (slot duration) and binary variables. This branching strategy aims to tighten the linearization constraints and reduce the integrality gap, which otherwise increases computational effort. The algorithm showed improved performance over standard MILP solvers in some small-scale scheduling test cases. Despite these improvements, the computational burden for solving even small problems remained significant.

Zhang & Sargent (1996) extended the concept of RTN by Schilling & Pantelides (1996) to develop a general MINLP formulation capable of managing complex processes with flexible resource interactions. This was done by allowing discrete resource interactions at multiple points during batch task execution, not just at the start and end, as in Schilling & Pantelides (1996) model. This formulation also facilitated the optimization of batch sizes by handling tasks with variable extents more effectively. Despite its advantages, the resulting nonconvex MINLP posed significant computational challenges, and the authors acknowledged that no existing algorithm could guarantee a global optimum. They suggested that local optimization would provide satisfactory solutions.

Ierapetritou & Floudas (1998) presented a continuous-time formulation that decoupled task events from unit events, resulting in a smaller model with fewer binary variables compared to

earlier models by Kondili et al. (1993) and Schilling & Pantelides (1996). This decoupling eliminated the need for binary variables to represent the start of task i in unit j at point n, a common feature in previous formulations, and avoided nonlinear terms. Although a large number of continuous variables were necessary for complex problems, the proposed approach still produced smaller models overall. It surpassed previous methods in terms of model size, solution speed, and result quality. This reduction in model size, particularly in binary variables, led to faster solution times and improved computational performance.

Majozi & Zhu (2001) developed a continuous-time MILP formulation using SSN representation, which focuses solely on states and eliminates the need for task and unit binary variables required by earlier STN representations. They also introduced the concept of unit aggregation to further reduce binary variables in large-scale problems. This was done by treating multiple units within a stage as a single entity, provided the units have identical performance and the processes within the stage operate in phase. The approach significantly reduces the number of binary variables and constraints, resulting in shorter CPU times for solving scheduling problems. Although linearizing bilinear terms in duration constraints increases the number of constraints, potentially offsetting some computational gains from reduced binary variables, the SSN-based method still requires fewer binary variables and achieves similar or better performance compared to previous formulations. This makes it particularly advantageous for larger, more complex scheduling problems in industrial applications.

Sundaramoorthy & Karimi (2005) explored the use of synchronous slots, which employ fewer binary and constraint variables without decoupling tasks from units, as done by Ierapetritou & Floudas (1998), and avoids big-M constraints. They demonstrated that their approach is simpler, more efficient, and potentially tighter than the leading models in literature. Susarla et al. (2010) modified this work to use unit slots instead of process slots, allowing for a more flexible approach to handling shared resources in multi-grid models. Additionally, Susarla et al. (2010) incorporated various storage configurations and wait policies for material states, enhancing the model's versatility. This work highlighted the limitations of existing and proposed multi-grid approaches for managing shared resources.

Seid & Majozi (2012) later enhanced the formulation of Majozi & Zhu (2001) using unit-specific time points, which facilitated non-simultaneous material transfer and improved schedules compared to earlier methods. Their approach represented tasks as states to be processed rather than as input or output materials, as Majozi & Zhu (2001) had done resulting in smaller problem sizes and fewer binary variables due to the SSN framework. Their models required fewer time points and showed superior computational performance over single-grid and multi-grid continuous time models. A significant advantage was their accurate handling of FIS constraints, unlike the previous models by Ierapetritou & Floudas (1998) and Majozi & Zhu (2001), which failed to address this, leading to UIS behavior in the final schedules.

Vooradi & Shaik (2013) also improved upon previous approaches by implementing conditional sequencing of production and consumption tasks in a rigorous unit-specific event-based model. This allowed for more efficient handling of non-simultaneous material transfers and different unit-wait policies. They introduced four-index variables to track material flow between tasks, which reduced the number of events needed for optimal solutions. In 2017, the authors further refined this model by allowing material transfers between production and consumption tasks to occur at the same event, thereby decreasing the number of events required for optimal solutions (Shaik & Vooradi, 2017). This updated model also utilized four-index variables to track material flow from each producing task, further minimizing the required events. Overall, the latter model achieved a reduction of 2-3 events as compared to the former model for benchmark problems, often improving computational performance.

In much recent contributions, Mostafaei & Harjunkoski (2020) presented a MILP model that incorporates both single-grid and multi-grid approaches. This model, derived from generalized disjunctive programming, offers a compact formulation with robust linear relaxations. A notable advantage of the multi-grid approach is its capacity to produce optimal solutions with fewer event points than single-grid models, thereby enhancing computational efficiency. While the authors thoroughly compare their model with existing ones, further exploration of more complex scenarios or larger-scale problems could better illustrate the model's scalability and robustness.

Li et al. (2022) proposed two novel unit-specific, event-based MILP models for the short-term scheduling of multipurpose batch plants. These formulations explicitly incorporate the

concepts of direct and indirect material transfer, which allows for more rigorous sequencing and alignment of tasks across different units. Importantly, when storage capacity is constrained, the models allow partial batch transfers to downstream units and permit batch retention within processing units over multiple event intervals (capabilities that were absent in preceding approaches). This flexibility leads to a reduction in the number of required event points to achieve optimality, can result in improved objective values (up to a 67% profit increase) and reduced computational effort compared to previous models. Although the study primarily emphasizes advantages, a potential drawback that is not explicitly stated but implied by the complexity of such models, could be the increased difficulty in initial implementation or understanding for those unfamiliar with detailed MILP formulations. Nevertheless, the models demonstrate superior performance compared to existing methods, offering a more computationally efficient framework for batch scheduling.

The papers mentioned above highlight progressive advancements in production scheduling optimization, addressing the computational limitations of early MILP formulations and introducing more efficient continuous-time and hybrid models to tackle industrial-scale batch scheduling challenges better. Each study made strides in reducing computational complexity while enhancing model flexibility and solution speed. Collectively, these advancements refined batch scheduling by minimizing computational complexity, enhancing flexibility, and improving solution speed. This dissertation leverages mathematical techniques to address the scheduling problem of a multistage multiproduct textile dyeing facility. It uses the unit-specific event-based approach for its compact and flexible formulation of scheduling problems, thereby enhancing the problem size reduction and solution efficiency of the model.

2.3.2 Scheduling of Multistage Batch Processes

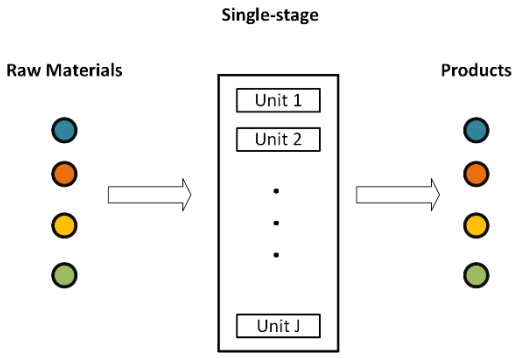
Over the past three decades, the scheduling of single- and multi-stage batch processes with multiple parallel units has gained significant attention due to the increasing complexity and demand for efficiency in various industries, such as chemical, textile, pharmaceutical, and food processing. The inherent challenge lies in the coordination and optimization of shared resources across multiple production units and stages while adhering to operational constraints such as timing, resource availability, and equipment limitations. In single-stage batch processes, all tasks are processed in a single stage, frequently utilizing multiple parallel units. The scheduling in such processes mainly focuses on minimizing idle times and optimizing the order in which

batches are processed to maximize throughput and reduce operational costs. Single-stage processes are typically less complex than their multi-stage counterparts, as they do not involve intricate dependencies between different stages of production. However, the inclusion of multiple parallel units introduces a level of complexity in terms of resource allocation, ensuring optimal use of all units without causing delays or inefficiencies. As the complexity of processes has grown, early models based on MILP, such as those by Kondili et al. (1993), have been instrumental in providing structured frameworks for optimizing single-stage operations.

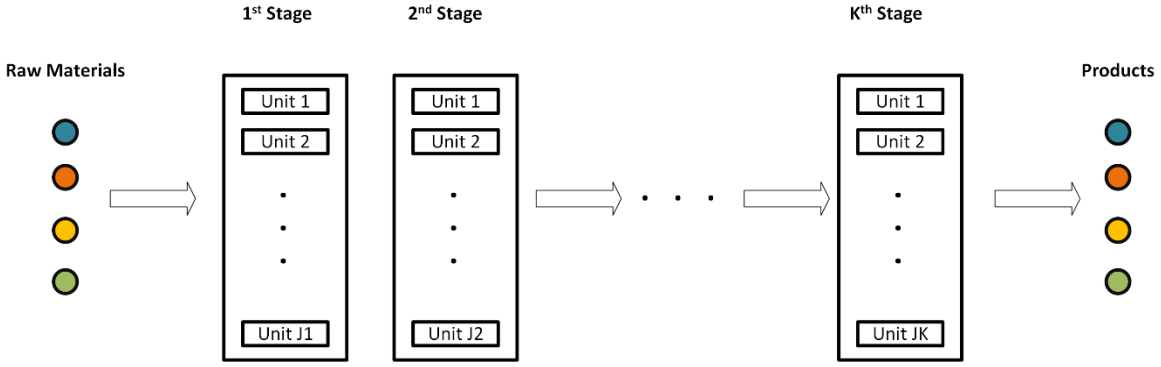
Cerdá et al. (1997) developed the first MILP model for scheduling single-stage multiproduct batch plants with parallel lines. The model determines the optimal allocation of jobs to units, job sequences, and start/completion times to minimize objectives like total tardiness, makespan, or number of tardy jobs. To improve computational performance for large-scale problems, heuristic rules are used to reduce the set of feasible predecessors for each order, reducing model size while still allowing flexibility in job assignment and sequencing. This model is, however, limited to single-stage processes and does not address the more complex multi-stage batch plants, which are more prevalent in the process industry.

In the multi-stage batch processes, processing units handle multiple batches. These batches, composed of one or more raw materials, are combined before the process begins and pass through several predefined stages, as illustrated in Figure 2.9. Scheduling these processes becomes exponentially more complex due to the need for specific tasks to be performed in parallel, with each stage potentially dependent on the completion of tasks in previous stages. Additionally, the presence of parallel units at each stage further complicates scheduling, requiring effective task allocation to avoid bottlenecks.

Pinto & Grossmann (1995) introduced a continuous-time MILP model for scheduling multistage batch plants with parallel equipment. This model used parallel time coordinates for both processing units and production orders, which provided a more flexible approach to scheduling by eliminating the need for time discretization. The authors also included preordering constraints and developed a decomposition algorithm that significantly improved the computational efficiency of the model, rendering it suitable for solving large-scale scheduling problems. However, the performance of the model declined with increasing problem size, particularly in scenarios involving more than 25 orders and 5 stages.



(a) Single-stage



(b) Multi-stage

Figure 2.9. A general depiction of single-stage and multi-stage processes

Harjunkoski & Grossmann (2002) then presented two strategies to mitigate combinatorial complexity in single-stage and multistage optimization scheduling problems involving cost minimization and due dates. These strategies integrate MILP and constraint programming (CP) methods, offering flexibility by allowing either MILP or CP to be used for sequencing based on the characteristics of the problem. However, the MILP model faces a combinatorial explosion due to the large number of binary variables, particularly in multistage problems.

Castro & Grossmann (2005) presented a novel continuous-time formulation that extends a single-stage formulation to accommodate multiple stages by introducing transfer time variables and constraints to represent interactions between stages. This formulation uses multiple time grids, one for each equipment resource, enabling efficient schedule representation. The authors compare this new approach to existing methods, including discrete-time and continuous-time formulations, as well as CP. Performance was evaluated on 30 test problems across three objectives: total cost, earliness, and makespan minimization. A key finding is that no single

method is optimal for all problem types and objectives. The discrete-time formulation performed particularly well for earliness minimization, and a new algorithm was proposed for makespan minimization within this model. The CP approach excelled in makespan minimization but scaled poorly.

Castro et al. (2006) extended their previous study by introducing two multiple-time grid continuous-time models for scheduling multistage batch plants with sequence-dependent changeovers. The key innovation lies in the handling of changeover tasks, either explicitly or implicitly, within the model constraints. A notable feature is the integration of processing and changeover tasks into a single set, which reduces the number of model variables and enhances solution efficiency. The explicit approach provides more versatility in objective functions, while the implicit approach results in smaller models. As in Castro & Grossmann (2005), the study compares these new models with other MILP and CP approaches, demonstrating their superiority, particularly for total cost minimization. Overall, the multiple-time grid models proved better suited for single-stage problems, whereas the CP model excelled in makespan minimization.

Ferrer-Nadal et al. (2008) investigated the importance of transfer times in multistage batch processes, expanding on prior research, including the work of Castro & Grossmann (2005), to address the shortcomings of existing scheduling models. The authors highlighted that overlooking transfer times between processing stages can lead to infeasible schedules, especially in complex plant configurations with shared units and bidirectional product flows. They demonstrated that while simplifications are often necessary to reduce problem complexity, neglecting transfer times can compromise schedule feasibility in multistage processes. The study revisits existing mathematical models for batch scheduling and proposes new approaches to incorporate non-zero transfer times, ensuring proper synchronization between consecutive processing stages. Although these proposed approaches do not significantly increase computational effort, as they do not require additional variables or constraints, the tradeoff is that the model may yield higher makespan values compared to simplified formulations that neglect transfer time.

Prasad & Maravelias (2008) focused on batch sizing in their pioneering work on a MILP precedence-based formulation for the simultaneous batching and scheduling of multistage

batch processes. Unlike previous approaches that assumed fixed batch sizes, their formulation optimizes both the number and size of batches to meet orders. Through various examples, the authors demonstrated that their simultaneous approach outperforms sequential methods in achieving better time-related objectives and handling excess production. This approach can accommodate parallel units with different capacities, allowing for more flexible batch sizing. Additionally, it addresses a wider range of objective functions, including profit maximization, which previous formulations could not address. However, there are some limitations. The authors assumed constant processing times regardless of batch sizes, which may not always be realistic. The formulation also assumes unlimited intermediate storage and no utility requirements, potentially limiting its applicability in some real-world scenarios. Finally, it does not account for multiple orders of the same product.

Sundaramoorthy & Maravelias (2008b) expanded on the work of Prasad & Maravelias (2008) by incorporating variable processing times based on batch size. They addressed this limitation by modelling processing times as functions of selection and size variables, allowing for a more realistic representation of process dynamics, especially when batching is part of the optimization. Furthermore, the proposed model showed improved performance through the use of a global-sequence formulation, which was generally more effective than slot-based or local-sequence approaches. Recent extensions of this approach also include additional constraints to handle intermediate storage (Sundaramoorthy & Maravelias, 2008a).

Marchetti et al. (2012) presented a group-based approach for scheduling large-scale multistage batch processes handling multiple orders per product, addressing a gap not covered by Prasad & Maravelias (2008) or previous studies. This method reduces model complexity by clustering batches for each order into groups that follow the same manufacturing route. The flexibility in adjusting the number of groups strikes a balance between solution quality and computational efficiency. The model can handle multiple customer orders per product with different due dates, variable processing times, and sequence-dependent changeovers. Various sequencing constraints were tested, and examples with up to 92 batches were successfully solved. The group-based model allows controlling problem size by adjusting the number of groups per order, and even with fewer groups, it can quickly find good feasible solutions. However, using fewer groups may lead to suboptimal solutions, as it reduces the granularity of the model.

Merchan et al. (2016) leveraged the strengths of the discrete-time formulation in managing various scheduling features and constraints, such as utility constraints and product diversification, and its potential for developing advanced solution methods. They developed four discrete-time formulations based on different representations of the scheduling problem and proposed several solution strategies, including tightening constraints, reformulations, and a solution refinement approach. Testing these models on various objectives, including earliness, makespan, and cost minimization, revealed that discrete-time models work well for earliness minimization. At the same time, combination of methods are necessary for makespan and cost objectives. Overall, the proposed approaches significantly outperformed existing continuous-time models, in some cases reducing computational time by up to four orders of magnitude. However, as with all discrete-time models, there is a risk of reduced accuracy since events are restricted to occurring at predefined time points. Additionally, these models tend to be larger than their continuous-time counterparts, resulting in higher computational demands.

Lee & Maravelias (2018) leveraged the flexibility of discrete-time models and the precision of continuous-time formulations to propose a hybrid approach. This method begins with a discrete-time model and then maps the solution onto material- and unit-specific continuous-time grids to establish sequencing relationships. A final continuous-time model refines the solution to obtain an accurate schedule. This approach retains the flexibility of discrete-time models while addressing their accuracy limitations. Demonstrated on several benchmark problems from the literature, the method showed strong performance in both solution quality and computational speed compared to previous approaches. The proposed model obtains high-quality solutions much faster than previous methods, often finding optimal solutions in seconds, whereas other models fail to converge in hours. Although the model may struggle with some features that continuous-time models handle more easily, its overall versatility and speed advantages make it highly practical for many applications.

Ackermann et al. (2022) presented a discrete-time MILP model for optimizing batching and scheduling in multiproduct multistage batch plant by introducing order consolidation. This strategy involves aggregating all individual customer orders for a particular product into a single, unified demand which results in the model reducing the number of required batches and enhances plant capacity utilization. Results from four examples demonstrate significant improvements: for instance, consolidating orders in one of the examples reduced batch

requirements from four to three, while in another achieved a 20% reduction in. The proposed approach effectively addresses small orders, ensuring timely fulfilment before due dates, and showcases robust computational performance across various scenarios. However, the complexity arising from order consolidation can result in significant computational effort and longer solution times, particularly for larger instances.

2.3.3 Scheduling of Multistage Multiproduct Batch Processes

The scheduling of multistage multiproduct batch processes presents a unique set of challenges, particularly when it comes to optimizing production while meeting varying demands for multiple products. These processes require coordinating resources and equipment across several stages, each with distinct processing times and material flows. The complexity increases when multiple orders for a product are considered, necessitating more binary variables in the MILP formulation, which significantly expands the problem size and computational effort. These binary variables are crucial for decision-making, such as resource allocation, task sequencing, and batch operation timing. However, the resulting large-scale optimization problems are computationally intensive, especially as the number of products and processing stages increases, making effective scheduling a critical research area in process systems engineering. This complexity necessitates advanced optimization methods that balance efficiency with practical computation times, as emphasized in studies by Kondili et al. (1993) and Pinto & Grossmann (1995). The following is a review of key studies over the years in multistage multiproduct scheduling, showcasing the evolution of approaches and methodologies developed to address the increasing complexity of these batch processes.

Erdirik-Dogan & Grossmann (2007) presented a continuous-time MILP model that uses time slots to represent potential batches, incorporating mass balances and changeovers. To address computational challenges, they developed a bilevel decomposition algorithm that splits the problem into upper-level sequencing and lower-level scheduling sub-problems. The upper level provides tight bounds and accurate predictions, while the lower level solves a simplified version of the original problem. Symmetry-breaking constraints are included to reduce redundant solutions and improve efficiency. The approach, demonstrated through several examples, shows significant computational benefits over full-space methods for larger problems. However, the formulation might exhibit subcycles which, while providing a valid upper bound on profit, prevent obtaining an actual sequence. Furthermore, similar to discrete-

time models, the limitation of slots to specific time periods could lead to suboptimal solutions, as slots cannot extend into other periods.

Liu & Karimi (2007) proposed several MILP formulations for scheduling multistage multiproduct batch plants with nonidentical parallel units and unlimited intermediate storage. They developed 11 new models using 4-index and 3-index binary variables, aiming to enhance existing formulations. Their models generally provide better initial RMIP values and solve faster than previous models, demonstrating flexibility in handling various objectives beyond makespan minimization. The study explores different modelling approaches, including unit-slots and stage-slots. Despite developing 11 formulations, the scheduling problem remains challenging, and a key finding is that no single model consistently performs best across all scenarios, highlighting the value of using multiple competitive models simultaneously.

Castro & Novais (2009) accounted for multiple product batches and sequence-dependant changeovers, which Prasad & Maravelias (2008) did not explicitly consider. The model uses 4-index binary variables and builds on previous work related to single-batch and multiple-batch problems without changeovers. Compared to other approaches, the new formulation is shown to be tighter and more computationally efficient, typically requiring fewer event points to find the global optimum. However, as with other multistage multiproduct scheduling problems, the complexity increases with the number of products and stages, potentially limiting scalability for very large problems.

Zeballos et al. (2011) proposed a CP formulation consisting of both a model and search strategies to handle various industrial features such as finite unit ready times, dissimilar parallel equipment, sequence-dependent changeovers, and different interstage storage policies. This approach was rigorously tested on numerous examples with varying difficulty levels and interstage storage policies. It performed well for different objective functions, including makespan, tardiness, and earliness minimization. However, the CP approach showed sensitivity to the number of orders, achieving optimal solutions for all 5-order examples but suboptimal ones for most 22-order problems. The performance of the model was also influenced by the chosen interstage storage policy, with UIS requiring more computational effort. Additionally, the approach struggled with larger problems, failing to achieve optimal solutions for several cases within the time limit.

Novara et al. (2016) also used the CP approach but focused on larger-scale problems and considered multiple batches per order in a campaign operating mode. Unlike Zeballos et al. (2011), this model can handle resource limitations for both processing and cleaning activities, which significantly affect solution feasibility. Additionally, this work demonstrates superior computational performance, finding high-quality solutions quickly for complex instances. The approach also shows greater flexibility in addressing various objective functions and storage policies. While the model performs well, optimality can only be guaranteed for a limited number of examples, which may be a limitation for some applications.

Shi et al. (2017) introduced an enhanced scheduling model for multi-stage, multi-product batch plants with parallel units. The approach divides the production process into sequential single stages using concepts of 'virtual due date' and 'virtual order release time'. It uses forward or backward order assignment strategies, along with heuristic rules, to arrange order production based on different scheduling objectives. The effectiveness of the model is demonstrated through benchmark cases, achieving optimal solutions for small problems and improved results for larger problems compared to previous methods. However, the model assumes unlimited intermediate storage between stages, which may not be realistic for all production environments. Additionally, the method may face challenges with very large-scale problems due to the significant computational complexity involved in optimization.

Han et al. (2022) proposed a hybrid discrete differential evolution algorithm (HDDE) to address large-scale multi-stage multi-product batch plant scheduling problems. This algorithm uses a novel two-line coding scheme, using integer variables for unit assignment and continuous variables for order sequencing. To enhance search efficiency, two modified mutation operations and two local permutation strategies are incorporated. The HDDE algorithm was tested on several large-scale problems and compared with MILP methods and other meta-heuristic algorithms. Results indicated that HDDE significantly outperforms other approaches in terms of solution quality and computational time. However, the HDDE algorithm has some limitations. It requires more CPU time per iteration compared to other algorithms, especially as the number of orders increases. Additionally, it may occasionally get stuck in local optima, showing some instability in reaching global solutions which makes it less efficient than MILP-based methods for small-scale optimal scheduling problems. Furthermore,

the complexity of HDDE increases when considering real production batch plant scheduling constraints.

The evolution of scheduling techniques has advanced from early MILP-based models, which emphasized binary decision-making and resource allocation, to more recent continuous-time formulations and CP approaches. This progression has aimed at achieving greater computational efficiency and scalability. Despite the development of innovative methods, significant challenges persist, especially in managing large-scale problems and multi-objective functions.

2.4 Water Network Synthesis (WNS)

The synthesis of water networks (WN) has emerged as a critical area of research in process systems engineering, particularly in the context of minimizing freshwater consumption and wastewater generation in industrial processes. The core elements of WNs typically include sources (water-generating processes), sinks (water-using processes), wastewater disposal sites, and regenerators (water treatment processes). The streams entering and exiting these components are interconnected using mixers and splitters. The objective is to optimize these networks for water reuse, recycling, and regeneration, often under constraints such as contaminant concentration limits and environmental regulations for wastewater disposal. The contaminant concentration in sources is generally known but varies for sinks, usually constrained by the maximum concentration the sink can handle. The contaminant concentration in wastewater disposal sites also varies but must stay within established environmental disposal limits. Additionally, it is common to assume a fixed quantity of available freshwater sources with a known unit cost for freshwater.

In WNS, a key distinction lies between systems that incorporate storage tanks and those that do not, as this choice significantly impacts water reuse strategies. Direct reuse involves immediately redirecting water streams from one process to another or recycling them within the same process, as illustrated in Figure 2.10. While efficient for processes with synchronized water demands, this approach limits operational flexibility. In systems without storage tanks, timing mismatches can arise when processes do not produce and consume water at the same intervals, therefore limiting water-saving potentials.

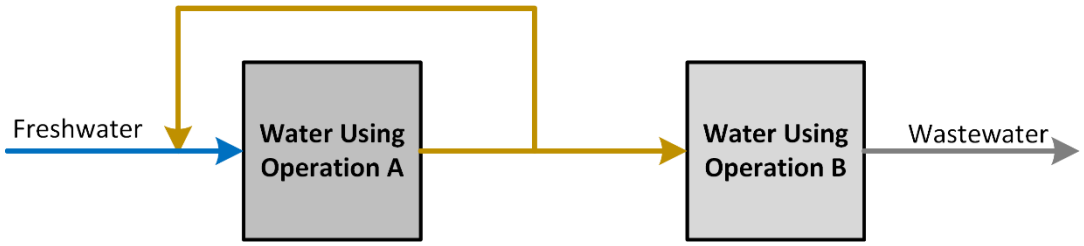


Figure 2.10. Direct reuse/recycle of water from operation A to B

On the other hand, indirect reuse involves the use of intermediate storage tanks, as illustrated in Figure 2.11, which add complexity to the WN but provide greater operational flexibility. By storing water between processes, indirect reuse allows the system to accommodate mismatches in water demand and supply, enabling more efficient management of water streams in multi-stage batch operations. The storage tanks serve as buffers, allowing water to be treated, stored, and/or reused when needed.

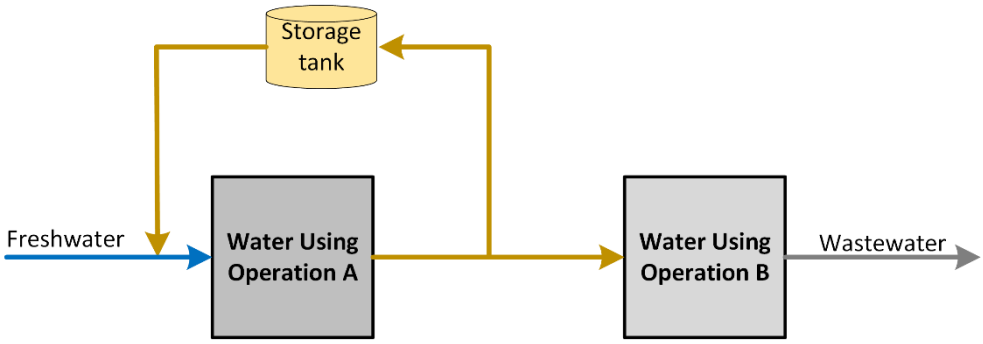


Figure 2.11. Indirect reuse/recycle of water from operation A to B

Over the years, a variety of insight-based and mathematical modelling techniques have been developed to tackle the complex nature of WN design and optimization. Insight-based approaches, such as water pinch analysis and mathematical modelling methods, including MILP and MINLP, have been extensively used to address the multi-objective, nonlinear, and nonconvex characteristics of WNs. Each method has its strengths depending on the characteristics of the problem. For instance, water pinch analysis provides valuable insights into the optimal reuse of water within the network by balancing contaminant load and water

availability without necessarily requiring extensive computational resources. Conversely, mathematical modelling techniques like MILP and MINLP are better suited for large-scale problems that require detailed optimization, such as determining the optimal allocation of water sources and sinks, incorporating regeneration options, and minimizing freshwater intake and wastewater discharge.

Heuristic methods have also found application in WN design, often used to generate near-optimal solutions within acceptable time frames when exact methods face computational limitations. These techniques are particularly useful when the non-linearity and non-convexity of the system render traditional optimization methods impractical. Additionally, as environmental regulations grow stricter and industrial water usage becomes more scrutinized, multi-objective optimization techniques have been increasingly incorporated into these models to balance conflicting goals such as makespan minimization, cost minimization, and freshwater reduction.

2.4.1 Insight-Based Methods for Water Network Synthesis

In WNS, a unified framework (either graphical or algebraic) is used to match individual water demand (sink) with appropriate supplies (source) (Buabeng-Baidoo et al., 2017a). Water pinch analysis (WPA) is commonly used as a graphical technique in WNS. It focuses on identifying the minimum freshwater requirements by constructing a composite curve which illustrates water usage within the system. Water pinch technology encompasses three primary tasks: (i) identifying water-consuming and effluent-generating activities, (ii) designing a water utilisation network to fulfil the objectives of these sources and sinks, and (iii) modifying the existing water utilisation network to optimise direct reuse and minimise effluent generation through effective process alterations (De Souza et al., 2011).

The earliest documented attempt at insight-based techniques on batch processes was a graphical-based method performed by Wang & Smith (1995). Their work used water pinch analysis in an attempt to maximise the concentration driving force, with time as the main process constraint. The targeting procedure involves dividing the problem into concentration and time intervals, with boundaries set by process endpoints. Water streams available for reuse are grouped by time intervals and reused within their concentration intervals. Surplus water is reused in subsequent intervals or stored for later use, but not in lower intervals. Freshwater is

introduced only after all reuse possibilities are exhausted, and any remaining surplus is discharged as effluent. The main drawback of this formulation is that it is more suited for semi-continuous processes as it allows the direct reuse of water between intervals with overlapping active time intervals (Majozi, 2005b). Furthermore, although the method allows for the rapid determination of minimum water flow requirements, the subsequent network design procedure is quite complex.

Olesen et al. (1997) presented a simplified procedure for designing water networks that handle single contaminants, addressing the complexity issue by Wang & Smith (1995). This approach utilizes the Load Table and remaining problem analysis as key tools to guide the design process, allowing for efficient network development while meeting minimum water targets. Initially, operations are classified based on their inlet/outlet concentrations relative to the pinch. A Load Table is then developed, showing contaminant loads and minimum water flows for each operation. The design process begins by addressing simpler operations to reduce problem size. For operations that are capable of reusing water, the remaining problem analysis determines if fresh water can be safely used. Operations are then positioned, and flows are set based on the Load Table, with water recycling from the pinch provided if needed. This methodology maintains the ability to explore design assumptions and consequences quickly but provides a more straightforward and practical approach to water network design. However, it is primarily intended for systems with single contaminants, which may not be applicable to more complex industrial scenarios with multiple pollutants. Additionally, the procedure relies heavily on the Load Table and classification of operations, which could become cumbersome for very large or complex systems.

Majozi et al. (2006) critiqued Wang & Smith (1995) for allowing water reuse among concurrently active processes, which is impractical in true batch operations. In their work, they proposed a graphical technique aimed at conserving water in batch processes wherein water reuse or recycling is exclusively permitted at the beginning or end of the process. This method provides flexibility by allowing either concentration or time to be the primary constraint, allowing for optimization based on the characteristics of the problem. The problem is divided into intervals based on concentration and time, enabling the cascading of water reuse opportunities through these intervals. Furthermore, it explores the use of process vessels as inherent storage, demonstrated through an Inherent Storage Availability Diagram. However,

this approach is limited to batch processes with single contaminant streams, a common limitation of graphical techniques.

Chen & Lee (2008) presented a graphical technique for designing water-using networks in batch processes, aiming to minimize freshwater consumption and wastewater generation. This approach uses quantity-time diagrams to represent water demands and sources, allocating them along a time axis to determine the network. Storage tanks are utilized to enable water reuse between non-overlapping operations. Nevertheless, the approach may not be able to achieve the maximum water recovery under certain operational constraints, such as restrictions on recycling between specific operations. Furthermore, the graphical nature of the technique may become cumbersome for large-scale systems with many operations and complex time schedules. Similar to the method by Majozi et al. (2006), this technique is restricted to systems with a single key contaminant.

Graphical techniques offer the advantage of providing physical insights for the designer; however, they share a common limitation of not being able to handle multi contaminant systems effectively. Furthermore, the composite curve used in the targeting process occasionally yields imprecise outcomes due to oversimplifications or assumptions made during the process. Alternative insight-based approaches, such as algebraic algorithms, have thus been developed to overcome some of the limitations of graphical techniques by offering greater precision and flexibility in handling multi-contaminant systems. Algebraic methods express process relationships, constraints, and objectives using algebraic equations, which allows for more precise and systematic analysis. Additionally, they are more efficient and support integration with other software, such as process simulators and spreadsheet programs (Foo, 2016).

Foo et al. (2005) developed an algebraic technique for targeting the minimum water flows for batch systems. The technique was based on a time-dependent water cascade analysis (WCA) in which various network targets, including freshwater, wastewater flow and storage capacity targets, were initially set. The method then used a new time-water network diagram to design the maximum water recovery network and meet the established targets. Networks with and without storage tanks were separated and prevented from mixing water sources with varying impurity levels meaning that a large number of storage tanks would be required if there were

many water sources with different impurity concentrations. Although this approach achieved the same utility targets as Wang & Smith (1995), it was able to provide a more systematic approach for targeting and design. It set various network targets, including overall and interval-based utility, storage capacity, and water allocation, prior to network design. Additionally, this method can manage more intricate scenarios, such as systems with non-overlapping water sources at different time intervals. However, the method is constrained by the assumption that water storage tanks are always available for reuse, overlooking the trade-off between network complexity (number of storage tanks) and operating costs.

Yongjian et al. (2007) introduced an algebraic method based on time-dependent concentration interval analysis (CIA) to design water utilization systems in batch processes. This approach effectively identifies opportunities for water reuse and reduces freshwater consumption in both single and repeated batch systems involving both mass transfer-based and non-mass transfer-based operations. Additionally, the method can concurrently minimize freshwater usage and determine optimal storage tank capacities. However, its application to large-scale systems with numerous operations and time intervals is complex and time-consuming due to manual implementation. Moreover, it mainly addresses single-contaminant systems, limiting its use in more complex real-world industrial scenarios with multiple contaminants.

2.4.2 Mathematical Optimization Techniques for Water Network Synthesis

Although graphical and algebraic methods are effective and easy to use, a significant drawback in WNS is their inability to handle multiple contaminants simultaneously, limiting their applicability in complex systems. Additionally, as with scheduling, graphical methods become impractical as the number of variables increases, and algebraic techniques can be cumbersome to scale up. Consequently, mathematical optimization techniques are often preferred because they can manage multiple variables, constraints, and complex relationships, making them suitable for more realistic models. Moreover, mathematical programming allows for sensitivity analysis, which examines how changes in model parameters impact the optimal solution (Winston, 2004).

In process integration, optimization-based approaches often apply superstructure optimization, which allows various steps to be performed simultaneously to determine the global optimum (Jezowski, 2010). A superstructure represents all potential connections within the system under

consideration, meaning the optimal solution is always a subset of this superstructure. This approach is crucial because it provides a unified and flexible starting point, allowing the model to identify the most efficient configuration from among numerous possible alternatives. Typically, optimization-based approaches are formulated as MINLP models, incorporating constraints such as water and contaminant balances, allocation, capacity, and storage limits to accurately capture the complexities of the system (Li & Majozi, 2019). Through these constraints, the superstructure model rigorously defines feasible solutions, ensuring that the derived optimal configuration meets all operational and environmental requirements. In the field of batch WNS with mathematical programming, mathematical optimization techniques can be categorized into sequential (fixed) and simultaneous (flexible) approaches. The sequential approach assumes a predetermined schedule, where the timing of each event is known before synthesizing the optimal water network (Croft et al., 2021). The simultaneous approach, on the other hand, determines the process schedule and minimizes freshwater usage at the same time.

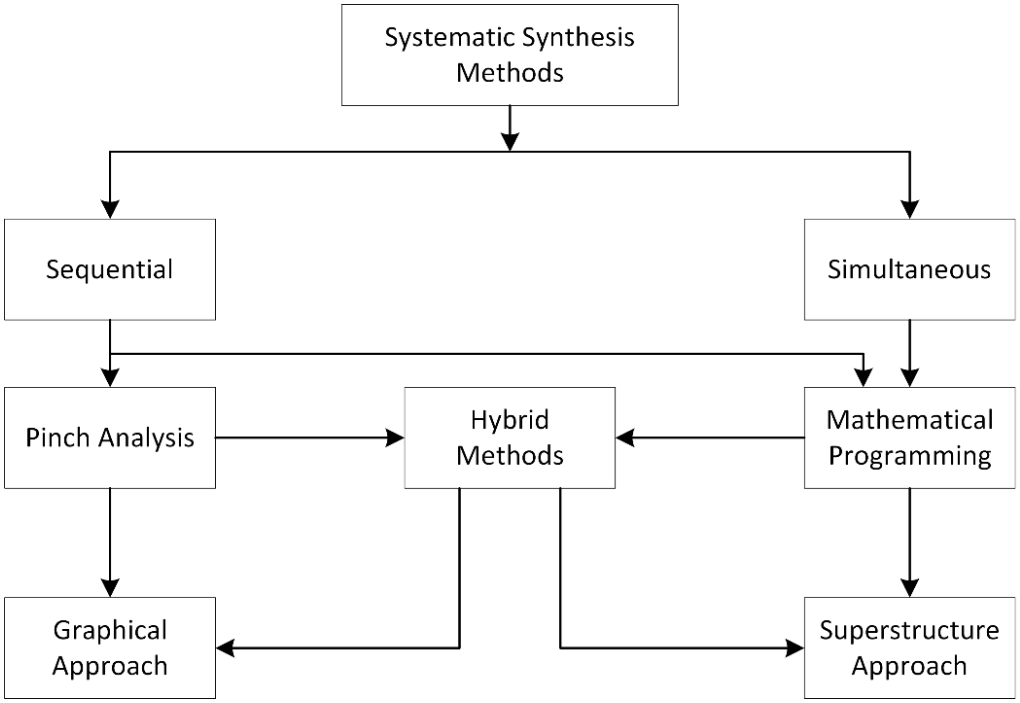


Figure 2.12. Classification of systematic synthesis methods

While sequential methods are easier to implement and yield good results, they overlook interactions between fresh resource flow rate and energy requirements, as well as the potential impacts on network design. Overall, they do not account for what is crucial in actually

implementing one of the recommended solutions in real process industries, which is the total cost (Ghazouani et al., 2015).

A common drawback of mathematical optimization-based techniques, especially for nonconvex or nonlinear problems, is their computational expense (Abass & Majozi, 2016). These models require significant time, memory, and computational power, making them resource-intensive and sometimes impractical for large-scale problems. However, they are often preferred because they can handle multiple contaminant streams, unlike graphical techniques, which are limited to single contaminant processes (Majozi, 2005b). The Water Network Model in this dissertation uses mathematical optimization techniques due to their ability to manage complex multipurpose water issues alongside multistage multiproduct production problems. Furthermore, it uses the simultaneous approach due to its advantage in optimizing interrelated variables, such as water reuse/ recycling and scheduling, within a single unified framework. This method ensures that the trade-offs between water usage and production efficiency are fully captured, facilitating an integrated solution that minimizes freshwater consumption and wastewater generation while maintaining production efficiency.

2.4.3 Wastewater Minimization in Multipurpose Batch Processes

Wastewater generated in multipurpose batch processes is often characterized by high concentrations of contaminants. One major issue in modelling these systems lies in their inherent complexity and variability. These processes involve diverse operations, with different products being manufactured in varying sequences and quantities. Such variability leads to fluctuating water demands and wastewater characteristics, which makes it difficult to develop a single, comprehensive model that accurately reflects the entire operation. Additionally, incorporating options for water reuse/ recycle and treatment into the model introduces further complexity, as these strategies significantly impact the overall water balance and contaminant loads. Below is a review of some formulations proposed over the years to address the challenges of wastewater minimization in multipurpose batch processes. Several advancements have been made in the field, and these have been extensively discussed in reviews by Bagajewicz (2000), Jezowski (2010) and Khor et al. (2014).

Almato et al. (1997) developed a mathematical model using storage tanks to manage the time dimension of processes, achieving significant freshwater savings. This approach demonstrated

the effectiveness of heuristics and mathematical optimization in batch processes. However, it missed additional water savings from reusing in-plant treated wastewater, and heuristics do not guarantee a globally optimal solution. Majozi (2005a) introduced a continuous-time technique for water minimization in multipurpose batch processes, focusing on a case with a fixed maximum contaminant outlet concentration. While the model achieved substantial water reductions, additional wastewater could not be recycled due to timing misalignments with other operations.

Majozi (2005b) later presented a continuous-time formulation that simultaneously optimized freshwater consumption and wastewater generation, with or without using a central reusable water storage facility. This flexible schedule technique was configured such that it exploits water reuse and recycle opportunities within one or more processes with a common single contaminant in the system. A key advantage of this formulation was its adaptability to various objective functions, allowing it to address multiple scenarios. However, the variables considered were either fixed or restricted to predefined bounds. Hence, Gouws et al. (2008) extended this model by presenting a more flexible mass transfer model, which allowed outlet concentration and the freshwater requirement to vary simultaneously. This resulted in a nonconvex MINLP model. Although an applicable solution procedure was provided, the nonconvex nature of the overall mathematical model meant that global optimality could not be guaranteed (Gouws et al., 2008).

Cheng & Chang (2007) proposed a novel model formulation that integrates the optimization of a production schedule, water reuse subsystems, and water treatment subsystem into a single framework. This model, based on STN representation, resulted in a MINLP model. Structure constraints were used to minimize the number of storage constraints and pipe connections. The integrated approach demonstrated significant efficiency improvements. However, the study faced limitations related to wastewater treatment capacity constraints, which led to an increase in the freshwater for wastewater dilution when treatment capacities were limited.

Majozi & Gouws (2009) later advanced the model developed by Majozi (2005b) by extending it to handle multiple contaminant streams. Their methodology examined two scenarios: one with limited storage and one without. They used the novel solution procedure proposed by Gouws et al. (2008) to solve both scenarios, achieving significant water savings. A key

advantage of this methodology is its broader applicability, as it addresses wastewater with multiple contaminants, which is more common in industrial settings.

2.4.4 Wastewater Minimization in Textile Plants

Wastewater minimization in textile plants involves strategies to reduce water use, recycle water, and improve treatment efficiencies, thereby reducing the environmental footprint of production. Numerous studies have explored different innovative approaches, such as the reuse of process water, process modification, closed-loop systems, and advanced treatment technologies like membrane filtration and electrochemical processes. This review focuses solely on mathematical optimization approaches within the context of process integration proposed for the textile industry.

In earlier work, Ujang Z et al. (2002) designed a water network for an old textile plant using pinch analysis, achieving significant reductions in both freshwater consumption and wastewater discharge. However, while the results proved to be promising for the specific textile plant studied, applying WPA to other textile facilities, especially newer ones with different layouts and processes, might yield varying results. Another limitation is that the impact of water quality changes during reuse cycles were explicitly considered. Process water quality can change throughout various stages of textile production. The study would benefit from incorporating an analysis of how these changes might affect the feasibility and effectiveness of the proposed water reuse strategy. Additionally, due to its usage of straightforward heuristics, pinch analysis generally lacks rigor, hence not always yielding the most optimal solutions.

Zhou et al. (2017) applied genetic algorithm optimization for scheduling textile dyeing production with a focus on water conservation, using actual production data from a textile dyeing plant to enhance practical relevance. The study examined two scenarios: one without unexpected order inserts and one with them, providing insights into the performance of the optimization method under different conditions. As the study mainly focused on water consumption, it did not consider other crucial factors in production scheduling, such as production time. Further, assumptions like all orders taking 12 hours to complete and having only three-color depths may oversimplify the real-world complexity of textile dyeing operations. Although the authors mention the potential for reusing wastewater, they did not

integrate this directly into the optimization model, which could have further increased water savings.

Nema et al. (2021) proposed a simple and intuitive water-reuse strategy using a GA for a multistage, multiproduct batch process in the textile industry. This novel approach uses intermediate storage tanks which rely on a heuristic for mapping discharge water from specific stages to corresponding tanks, resulting in significant water savings. However, a major operational constraint was the timing mismatch between when a stage discharges water and when another stage could potentially reuse it. Additionally, the water reuse scheme sacrifices some potential freshwater savings by not allowing mixing with higher concentration discharges, prioritizing operational simplicity instead. Furthermore, the stochastic nature of GAs limits the ability of the model to find the optimal solution consistently, as they often converge to suboptimal solutions.

Croft et al. (2021) leveraged the advantages of discrete- and continuous-time models by combining the two models as in the work by Lee & Maravelias (2018). Their framework aimed to minimize freshwater consumption in textile plants through the sequential use of unit-specific and global continuous-time methods. The optimal schedule efficiently produced different shades in each unit, reducing idle time. The water schedule results showed significant reductions in freshwater usage and wastewater discharge, enhancing productivity and environmental efficiency. However, a common drawback of sequential approaches is their loss of generality compared to simultaneous optimization, potentially missing better solutions that consider schedule-water network tradeoffs. Additionally, the water minimization scenarios failed to converge to proven optimal solutions within the 12-hour computational limit, indicating limitations in solving large industrial problems to optimality.

Bhavsar et al. (2023) recently extended the work of Nema et al. (2021) by incorporating intentional waiting to the desynchrony in water reuse caused by mismatched water demand and availability across different stages. Their results showed a significant increase in freshwater savings with an increase in the makespan. One advantage of this strategy is that it allows for greater freshwater savings with fewer storage tanks compared to using storage tanks alone without introducing waiting periods between stages. However, the challenge of suboptimal solutions due to the use of GAs persists.

2.5 Challenges in Water Network Synthesis

2.5.1 Non-linearity and Non-convexity

Non-convex nonlinearity is a significant challenge in the synthesis of batch water networks (BWN) due to the complex interactions between unit water balances, contaminant concentrations, and operational constraints. As noted by (Bagajewicz, 2000), these complex interactions make it difficult to linearize the problem. These non-convex relationships often create a solution space with multiple local optima, complicating the search for the global optimum. Samsatli & Papageorgiou (2003) demonstrated thathe time-dependent nature of batch processes also exacerbates nonlinearity. In these systems, water usage patterns and contaminant loads vary over time, and scheduling decisions must be integrated with the synthesis of the water network. This time-dependent variability introduces additional nonlinear constraints, as the water demand, reuse, and recycle streams fluctuate dynamically over the different stages of the batch process. Optimizing these dynamic systems often results in non-convex formulations, further increasing the complexity of the problem and the computational burden of finding optimal solutions (Samsatli & Papageorgiou, 2003).

Wang & Smith (1995) pointed out that the use of MINLP is often necessary to handle both the discrete decisions and the continuous variables involved in WNS. However, solving MINLP models is computationally challenging due to the non-convex nature of the problem, which can lead to difficulties in finding global optima. Solving these models to global optimality typically requires advanced solution techniques (discussed in subsection 2.6.4), which can be computationally intensive and time-consuming. Researchers have developed approximation techniques, such as linearization (discussed in subsection 2.6.3), relaxation, or decomposition methods to manage the non-convexity. These methods aim to reduce the complexity of the non-convex model by breaking it down into smaller, more manageable sub-problems or by approximating the non-convex relationships using simpler convex representations. However, these approaches can lead to suboptimal solutions because the approximations may not fully capture the intricacies of the original non-convex problem.

2.5.2 Binary Variables

Binary variables typically represent discrete decisions, such as whether to reuse, treat, or dispose of specific water streams or whether certain equipment or treatment processes are

active at particular times. Handling binary variables in optimization requires the use of MILP or MINLP models (discussed in subsection 2.6.1), which are much more computationally intensive than purely continuous models. One of the primary challenges with binary variables is the significant increase in model size. As the number of process units, tasks, and contaminants increases, the number of binary variables grows exponentially, leading to larger and more difficult-to-solve models. Bagajewicz (2000) and Karuppiah & Grossmann (2008) noted that the combinatorial nature of binary decisions creates a vast search space, making it harder for optimization algorithms to find global or even high-quality solutions in a reasonable time frame.

Additionally, binary variables are involved in the activation or deactivation of water-using processes or treatment units in specific time intervals, which is especially critical in batch processes where timing and scheduling are key. For example, if a binary variable determines whether a treatment unit is used in a certain time period, the interactions between this variable and other process variables, such as contaminant concentrations, introduce nonlinearities into the problem. These nonlinearities make solving the MINLP model even more challenging, particularly because the use of binary variables often leads to a non-convex solution space.

2.5.3 Multi-Objective Optimization

Multi-objective optimization poses a significant challenge in the synthesis of BWN due to the need to balance conflicting objectives, such as minimizing water consumption, reducing operational costs, and minimizing production time. In such systems, trade-offs often exist between these objectives, making the optimization problem more complex. In water network design, one of the primary objectives is to minimize freshwater consumption, which directly affects both operational costs and environmental sustainability. However, optimizing wastewater generation and minimizing production makespan adds complexity, as these objectives often conflict with cost-saving measures. For instance, increasing water reuse/ recycling within a batch process can lower freshwater usage. However, it may raise capital and operational costs due to the need for additional equipment like storage tanks (Foo, 2009). Similarly, minimizing water usage without storage tanks can affect production time, as reusing water from time-misaligned streams can be time-consuming. Often, there is no single "optimal" solution that simultaneously minimizes all objectives. Instead, the goal is to find a set of *Pareto-optimal* solutions that represent a trade-off between different objectives.

2.6 Background on Mathematical Modelling

Mathematical modelling involves translating complex systems or processes into mathematical forms, allowing for simulation, optimization, and prediction. Mathematical models help in understanding system behavior, making decisions, and designing optimized solutions. They consist of mathematical relationships, including equations, inequalities, and logical expressions, that describe a physical situation or system. These models can be broadly classified into several types i.e., empirical, stochastic and deterministic models (Goode, 2004). Empirical methods are based on observation, experimentation, and direct experience rather than purely theoretical analysis. They rely on gathering data through experiments, surveys, or real-world measurements and then analyzing this data to form conclusions or predictions. Stochastic methods are methods that often rely on physical analogies in order to generate trial points that mimic the approach to an equilibrium condition (Grossmann & Biegler, 2004). They incorporate randomness and uncertainty, resulting in different possible outcomes even under identical conditions. Deterministic models precisely determine the outcome by the model parameters and initial conditions, with no randomness involved. In these models, each set of inputs consistently yields the same output, making them ideal for applications where precision, consistency, and reproducibility are crucial.

Optimization, also known as mathematical programming, is a branch of mathematics and operations research focused on finding the best possible solution for a problem given a set of constraints and objectives. Its primary goal is to improve the performance of the system it is applied to. Optimization is a highly effective method for systematically addressing process integration challenges (El-Halwagi, 2017). An optimization model seeks to determine the values of decision variables that optimize (maximize or minimize) an objective function while satisfying the given constraints (Winston, 2004). Decision variables can be continuous, discrete or binary. Continuous variables can take any value within a range, and discrete variables can only take specific integer values, and binary variables can only be 0 or 1, indicating existence or non-existence. The objective function, or performance index, is expressed in terms of design variables, and the goal of optimization is to find the variable values that yield the optimal value of the objective function. Therefore, a mathematical optimization model consists of an objective function and a set of constraints expressed in the form of equalities or inequalities. In general, a mathematical optimization model can be expressed in the following format (Edgar et al., 2001):

$$\textbf{Minimize or Maximize}_{x} \quad Z = z(x) \qquad \text{objective function} \qquad (2.1)$$

$$\textbf{Subject to} \quad : \; h(x) \geq 0 \qquad \text{inequality constraints} \qquad (2.2)$$

$$\textbf{and} \quad : g(x) = 0 \qquad \text{equality constraints} \qquad (2.3)$$

The optimization model seeks to find the decision variables, x, that optimize the objective function, Z, while ensuring the model adheres to the limits set by the inequality constraints, h, and the equality constraints, g. For many real-world problems, finding an analytical solution to the optimization problem is often challenging. Therefore, an iterative approach known as numerical optimization is typically used. The iterative process usually involves the optimizer testing the model with different values for the decision variables, x, as shown in Figure 2.13. In each iteration, the model uses these values to simulate the situation and calculates the objective function and the constraints. The optimizer then adjusts the values based on this feedback, gradually working toward the best solution. This cycle repeats until the optimizer satisfies the conditions necessary to find the optimal answer.

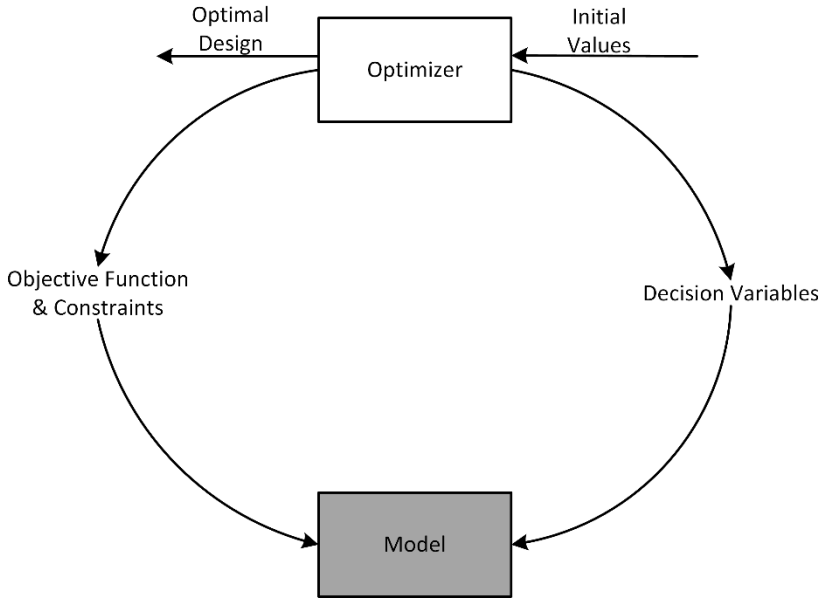


Figure 2.13. An illustration of the numerical optimization framework (Diwekar, 2020)

The general form in Equations 2.1 – 2.3 includes several constraints. When there are no constraints, it is known as an *unconstrained optimization* problem; otherwise, it is a *constrained*

optimization problem. Some optimization problems aim to find the global minimum, while others seek the best solution within a specific range. Constrained optimization focuses on finding the optimal value of the objective function within the limits set by the constraints on decision variables. In contrast, unconstrained optimization aims to find the global maximum or minimum of a function across its entire domain without any restrictions on the decision variables.

2.6.1 Classification of Optimization Problems

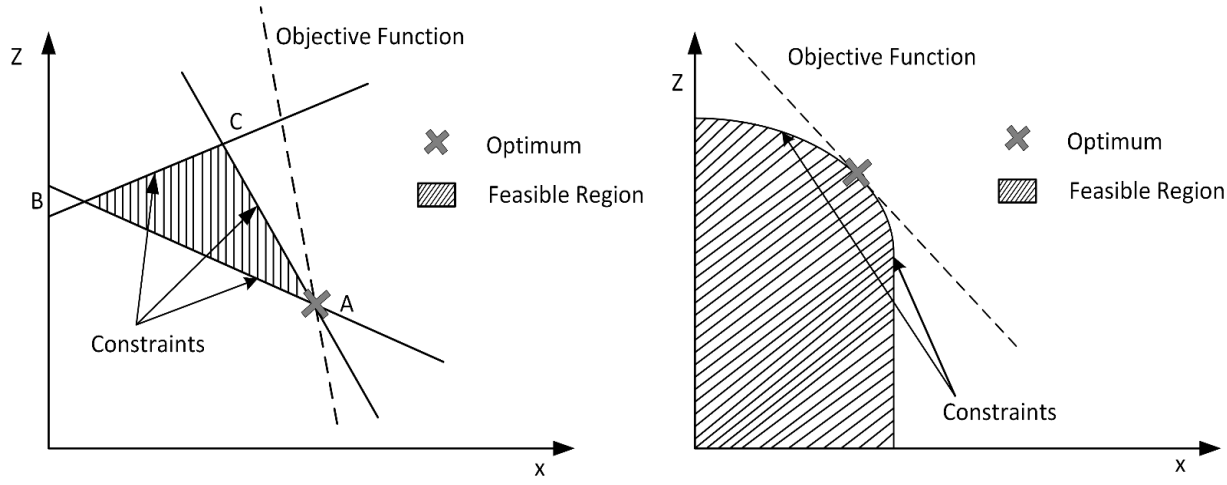
Optimization problems can be categorized based on the nature of their objective functions, constraints, and decision variables (Edgar et al., 2001). They are generally classified as either discrete or continuous, as shown in Table 2.1 below. In discrete optimization, decision variables can only assume specific, distinct values, typically integers or binary values (e.g., 0 or 1). This type of optimization is often used in scheduling, logistics, and resource allocation problems, where variables represent counts or binary decisions. In continuous optimization, decision variables can take any real number values within a specified range. This approach is commonly applied in fields like engineering, finance, and science, where variables do not need to be restricted to whole numbers. Mixed integer optimization is a type of optimization method that involves both integer and real variables in the objective function (Dutta, 2016). In continuous optimization, the complexity of a problem often depends on whether the objective function and constraints are smooth (differentiable).

Table 2.1. Classes of optimization problems (Dutta, 2016)

Optimization		
Continuous		Discrete
Differentiable	*Non-differentiable*	
Linear Programming (LP)	Direct Search	Integer Programming (IP)
Quadratic Programming (QP)	Stochastic Search	Mixed Integer Linear Programming (MILP)
Nonlinear Programming (NLP)		Mixed Integer Nonlinear Programming (MINLP)

Differentiability affects which solution methods can be used, as many algorithms rely on derivatives to navigate the solution space. Smooth functions have well-defined slopes, allowing optimization methods to use gradients for efficient navigation. However, nondifferentiable problems have sharp changes or discontinuities, making derivatives undefined at some points. For these cases, stochastic search methods like simulated annealing (SA) or genetic algorithms (GA) are useful. These methods do not require derivatives and use randomness and probabilistic rules to explore the search space, helping to avoid getting stuck in local optima.

Figure 2.14 (a) shows a linear programming (LP) problem where both the objective function and constraints (lines AB, BC, and CA) are linear, meaning ($z(x), h(x)\ and\ g(x)$) are all linear. LP problems fall under convex programming, where the objective function is convex (linear) and the constraints form a convex feasible region, ensuring that the optimal solution is a global optimum. If at least one constraint or the objective function is nonlinear, the model is referred to as Nonlinear Programming (NLP).



(a) LP Problem (b) NLP Problem

Figure 2.14. Linear and nonlinear programming problems (Diwekar, 2020)

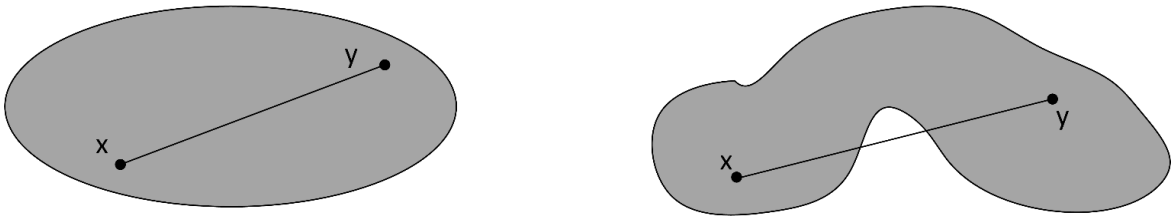
In both LP and NLP, the decision variables are scalar and continuous. However, when decision variables must be whole numbers, i.e., integers, such as for indivisible items like people, machines, or units of products, this is known as Integer Programming (IP) or, more commonly, Mixed Integer Programming (MIP). The decision variables in this programming class are scalars and integers. Depending on the nature of the objective function and constraints, MIPs can be further divided into MILP and MINLP. In MILP, both the objective function and

constraints are linear, whereas in MINLP, the problem involves both integer and continuous decision variables with at least one nonlinear component.

Optimization problems range from small-scale to large-scale problems. Small-scale problems have fewer than 100 constraints and 500 variables, medium-scale problems have fewer than 400 constraints and 1 000 variables, while large-scale problems can have up to 10^9 constraints and variables. Larger problems with more decision variables and constraints demand greater computational power and time to solve. As the number of variables increases, the solution space (the range of all possible solutions) expands exponentially, making it harder to explore thoroughly. The interaction of problem size, variable type, and nonlinearity further amplifies the complexity. Problems with numerous variables that are nonlinear and include integer constraints are particularly challenging, as they combine the difficulties of large-scale search, combinatorial complexity, and non-convexity.

2.6.2 Convexity vs. Non-convexity

NLP models can be classified as either convex programming or non-convex programming models. Convexity describes a set where, if you pick any two points inside it, the entire line connecting those points will also be inside the set. Conversely, the shaded area in Figure 2.15(b) represents a non-convex region because, although points *x* and *y* are within this region, the line segment connecting them does not lie entirely within the region.



(a) Convex set (b) Non-convex set

Figure 2.15. Illustration of convex and non-convex sets (Williams, 2013; Diwekar, 2020)

An optimization model is considered convex if it involves minimizing a convex function within a convex feasible region, meaning that any local optimal solution is also a global optimal solution (Williams, 2013). This property ensures that there is only one global optimum without any other local optima, as seen in LP problems. However, this is not the case for NLP and

MINLP problems, where non-convexity can lead to multiple local optima, making it challenging to identify the global optimum. In mathematical programming, convexity is valuable because it determines whether a single best solution exists or if the problem contains multiple local and global optima, which directly impacts the choice of solution methods and the difficulty of finding the optimal solution. Nonlinearities introduce additional complexity, as nonlinear problems are not always convex and can contain several local optima, as illustrated in Figure 2.16. In such cases, algorithms may become trapped in local optima if they fail to escape smaller peaks in the solution space, especially in non-convex functions with multiple suboptimal solutions.

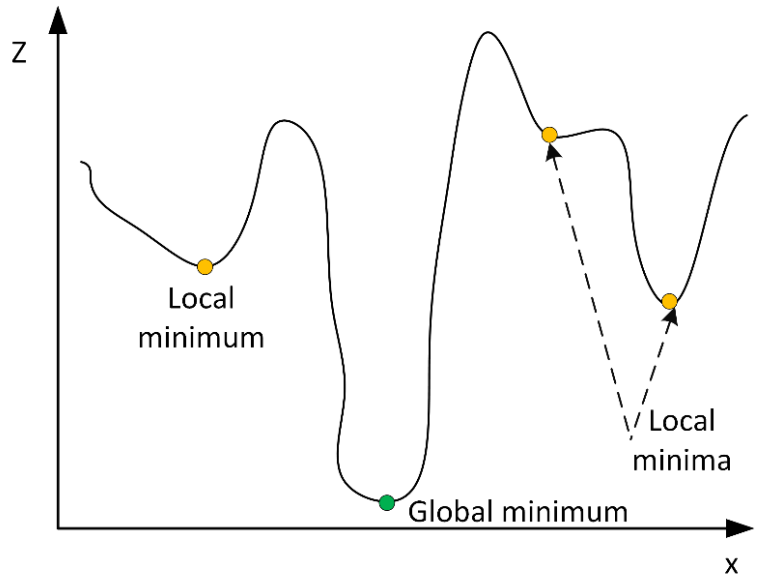


Figure 2.16. Local optima and global optimal solution in non-convex NLP problems

Convexification is used to reformulate a non-convex problem into a convex one to allow solvers to explore the entire solution space effectively without being restricted to local regions. Two widely used convexification techniques are McCormick envelopes (McCormick, 1976) and Glover Transformation (Glover, 1975), which linearize bilinear terms. The following subsection will discuss how bilinear terms can be reformulated into linear terms.

2.6.3 Linearization of Bilinear terms

Bilinear terms are mathematical expressions that involve the product of two variables. This section discusses the following product combinations, which are directly applicable to the

proposed model in this dissertation: the product of two continuous variables and the product of a binary variable and a continuous variable.

A. Product of two continuous variables

McCormick (1976) proposed a methodology to obtain global solutions for non-convex NLP problems, which involves the generation of convex and concave envelopes for factorable functions with bilinear terms. This method creates piecewise linear underestimators and overestimators to bound the non-convex function from below and above, respectively, forming a convex approximation of the original problem. The bilinear term in equation (2.4) is then replaced with these bounds, making the optimization problem convex and enabling solvers to efficiently explore the feasible region and avoid getting trapped in local optima.

Let z be the bilinear term, with continuous variables x and y

$$z = xy\,, \quad \forall x, y \in R \tag{2.4}$$

Each continuous variable x and y is bounded between lower and upper bounds as shown in constraints (2.5) and (2.6)

$$x^L \leq x \leq x^U \tag{2.5}$$
$$y^L \leq y \leq y^U \tag{2.6}$$

Constraints (2.5) and (2.6) are decomposed and rearranged to form constraints (2.7) - (2.10) below.

$$x - x^L \geq 0 \tag{2.7}$$
$$x^U - x \geq 0 \tag{2.8}$$
$$y - y^L \geq 0 \tag{2.9}$$
$$y^U - y \geq 0 \tag{2.10}$$

By taking the product of the different combinations in constraints (2.7) – (2.10), the following constraints are obtained.

$$xy - x^L y - xy^L + x^L y^L \geq 0 \tag{2.11}$$
$$xy - x^U y - xy^U + x^U y^U \geq 0 \tag{2.12}$$

$$-xy + xy^L + x^U y - x^U y^L \geq 0 \quad (2.13)$$

$$-xy + xy^U + x^L y - x^L y^U \geq 0 \quad (2.14)$$

The following McCormick (1976) underestimators (constraints (2.15) and (2.16)) and overestimators (constraints (2.17) and (2.18)) are obtained by substituting xy from equation (2.4) and rearranging the terms.

$$z \geq x^L y + xy^L - x^L y^L \quad (2.15)$$

$$z \geq x^U y + xy^U - x^U y^U \quad (2.16)$$

$$z \leq x^U y + xy^L - x^U y^L \quad (2.17)$$

$$z \leq xy^U + x^L y - x^L y^U \quad (2.18)$$

B. Product of binary and a continuous variable

Glover (1975) proposed a method used to linearize nonlinear terms in optimization problems, particularly those involving binary and continuous variables. This linear transformation preserves the exact nature of the original problem but reformulates it in a way that MILP solvers can handle efficiently.

Let z be the bilinear term, with continuous variable x and binary variable y

$$z = xy, \quad \forall x \in R, y \in [0,1] \quad (2.19)$$

The continuous variable x is bounded between lower and upper bounds as shown in constraint (2.22), and the binary variable y can take the value 0 or 1 such that:

$$z = 0 \quad \text{if } y = 0 \quad (2.20)$$

$$z = x \quad \text{if } y = 1 \quad (2.21)$$

$$x^L \leq x \leq x^U \quad (2.22)$$

Constraint (2.23) is obtained by multiplying constraint (2.22) with the binary variable *y,* then substituted with z to obtain constraint (2.24). Constraints (2.20) and (2.21) can be satisfied by employing the transformations shown in constraints (2.24) and (2.25).

$$x^L y \leq xy \leq x^U y \quad (2.23)$$

$$x^L y \leq z \leq x^U y \quad (2.24)$$

$$x - x^U(1-y) \leq z \leq x - x^L(1-y) \quad (2.25)$$

Equation (2.19) can be linearized by replacing it with constraints (2.24) and (2.25). This approach exactly linearizes and convexifies the model, necessitating the introduction of a new continuous variable and four additional constraints for each bilinear term, but it ensures a global optimum.

2.6.4 Solution Methods and Solvers for MINLP Problems

MINLP models are extensively used in optimizing chemical and industrial processes due to their ability to handle complex decision-making scenarios. As previously explained, these models incorporate nonlinear terms in the constraints and/or objective function and include both continuous and discrete (integer or binary) decision variables. MINLP problems are classified as NP-hard problems because of the dual nature of their variables, along with the presence of non-convexities and nonlinearity, which contribute to the computational complexity of their models (Bonami et al., 2011). Such problems inherit the combination of variables found in MIP models and nonlinearity and convexity challenges associated with NLP models (Boukouvala et al., 2016).

The complexity of MINLP models makes their solution computationally expensive, often requiring advanced algorithms and solvers. Several strategies can be employed to assist MINLP models in finding the global optimum. Among these techniques are linearization techniques, decomposition approaches, initialization point methods, and global optimization methods. Below is a brief overview of the different optimization techniques. This dissertation focuses on linearization techniques (discussed in the previous subsection) and global optimization methods.

- *Linearization techniques* - Linearization is used to convert nonlinear equations or terms in an optimization problem into linear approximations. This reformulation allows the problem to be solved using MILP solvers, which are computationally more efficient than solvers for nonlinear problems. Techniques such as McCormick envelopes (McCormick, 1976) and Glover Transformation (Glover, 1975) can be used.

- *Decomposition approaches* – This approach decomposes/ breaks the problem into smaller, more manageable sub-problems. It exploits the structure of the problem, such as the separability of variables, constraints, or components, and solves the sub-problems iteratively or hierarchically (Conejo et al., 2006). It uses techniques such as Lagrangian relaxation and Benders decomposition, amongst others.
- *Initialization point method* – These are strategies that select or generate a starting point for solving MINLP optimization problems. A good initialization can significantly improve solver performance by guiding it toward convergence and reducing computational effort, especially for non-convex problems that are susceptible to local optima. The starting point can be obtained from random initialization, heuristics-based, relaxation-based or prior solution-based initialization.
- *Global optimization methods* – Global optimization methods aim to find the absolute best solution (global optimum) to an optimization problem, ensuring that no better solution exists elsewhere in the solution space (Koop et al., 2024). They can be categorized as either stochastic or deterministic.

A. Stochastic Optimization

Stochastic optimization uses randomization and probabilistic exploration of the solution space to approximate the global optimum. The techniques can be broadly categorized into random search techniques, evolutionary methods, swarm intelligence methods, and others. Random involves sampling candidate solutions randomly within the defined search space and evaluating them based on an objective function. Examples include pure random search, simulated annealing (SA), and tabu search (Rangaiah, 2010). Evolutionary methods are algorithms inspired by biological evolution processes and include genetic algorithms (GA), differential evolution (DE) and evolution strategy (ES) (Tsai & Chang, 2001). The swarm intelligence methods are inspired by the collective behaviour of decentralized, self-organized systems found in nature, such as ant colonies, fish schools, and bee swarms. This group includes algorithms like particle swarm optimization (PSO), ant colony optimization (ACO), artificial bee colony (ABC) and chicken swarm optimization (Rangaiah, 2010; Meng et al., 2014; Maier et al., 2019).

Stochastic approaches are generally easier to design and implement, and they typically converge to a solution faster than deterministic methods (Koop et al., 2024). However, these

approaches are not rigorous, and their probabilistic nature can lead to suboptimal results, especially for large-scale problems.

B. Deterministic Optimization

Deterministic optimization uses systematic algorithms to explore the solution space, guaranteeing convergence to the global optimum within a finite time (Bonami et al., 2011). The commonly used algorithms in this approach include Branch-and-Bound (BB), Generalized Benders Decomposition (GBD), Outer Approximation (OA) and Extended Cutting Plane (ECP).

- *Branch-and-Bound* – the algorithm iteratively divides the solution space into smaller subregions (branches) and evaluates bounds to eliminate suboptimal regions (Boyd & Mattingley, 2010).
- *Generalized Benders Decomposition* – GBD splits a large problem into a master problem and sub-problems. The master problem deals with high-level decision variables, while the sub-problems focus on the details, such as specific decisions that depend on the master problem's choices (Geoffrmn & Balakrishnan, 1972).
- *Outer Approximation* - the OA algorithm decomposes MINLP problems into NLP and MILP sub-problems and iteratively approximates the feasible region to refine bounds (Duran & Grossmann, 1986).
- *Extended Cutting Plane* - ECP is an extension to the cutting plane method. It improves a solution by iteratively adding new constraints that help exclude poor solutions, gradually leading to the best one (Westerlund & Pettersson, 1995). It closely resembles the OA algorithm but differs in that it does not solve any NLP problems separately, as opposed to the OA method, which solves both the MILP problem and an NLP problem in the main iteration loop (Westerlund & Pettersson, 1995).

A variety of commercial and open-source solvers are available for handling MINLP problems. These solvers typically integrate multiple techniques and ideas from MILP and NLP subsolvers to enhance their performance (Boukouvala et al., 2016). They are implemented in various programming languages and are accessible either as standalone executables or as libraries within algebraic modelling software like GAMS (General Algebraic Modelling Systems), AMPL (A Mathematical Programming Language), and AIMMS (Advance Interactive

Multidimensional Modelling System) (Kronqvist et al., 2019). Additionally, some solvers are directly implemented in the same programming languages as their modelling systems, such as MATLAB, Python (Pyomo), and Julia (JuMP). Table 2.2 provides a summary of the commonly used MINLP solvers, their algorithms, and the platforms that they support.

Table 2.2. MINLP Solvers (Kronqvist et al., 2019)

Solver	Algorithm	Interface	Reference
AlphaECP (Alpha Extended Cutting Plane)	αECP (Alpha Extended cutting Plane)	GAMS, NEOS	Westerlund (2002)
ANTIGONE (Algorithms for coNTinuous / Integer Global Optimization of Nonlinear Equations)	BB or αBB (Branch-and-Bound)	GAMS, NEOS	Misener & Floudas (2014)
AOA (AIMMS Outer Approximation)	OA (Outer Approximation)	AIMMS	Hunting (2011)
BARON (Branch and Reduce Optimization Navigator)	BB (Branch-and-Bound)	AIMMS, AMPL, GAMS, JuMP, MATLAB, NEOS, Pyomo, YALMIP	Ryoo & Sahinidis (1996) and Tawarmalani & Sahinidis (2005)
BONMIN (Basic Open-source Nonlinear Mixed Integer Programming)	NLP-BB, LP/NLP-BB, OA, feasibility pump, OA-based branch and cut, and a hybrid approach	AMPL, C++, GAMS, JuMP, MATLAB, NEOS, OS, Pyomo, YALMIP	Bonami et al. (2008)
Couenne (Convex Over and Under Envelopes for Nonlinear Estimation)	LP-based Spatial BB	AMPL, C++, GAMS, JuMP, NEOS, OS, Pyomo	Belotti et al. (2009)
DICOPT (Discrete Continuous Optimizer)	OA	GAMS, NEOS	Viswanathan & Grossmann (1990)
Juniper	NLP-BB	JuMP	Kröger et al. (2018)
Knitro	LP/NLP-BB	AIMMS, AMPL, C++, C#, Fortran, Java, JuMP, GAMS, NEOS, Pyomo, Python, YALMIP	Byrd et al. (2006)

LINDO	Branch and Cut	C, C++, Delphi, Excel/What's Best!, Fortran, Java, JuMP, GAMS, LINGO, MATLAB, NEOS, .NET, Ox, Python, R	Lin & Schrage (2009)
MILANO (Mixed-Integer Linear and Nonlinear Optimizer)	NLP-BB and OA	MATLAB	Lee & Leyffer (2012)
Minotaur (Mixed-Integer Nonlinear Optimization Toolkit: Algorithms, Underestimators, and Relaxations)	NLP-BB and LP/NLP-BB	AMPL, C++	Mahajan et al. (2021)
Muriqui	ECP, ESH, OA, LP/NLP-BB, and NLP-BB	AMPL, C++	Melo et al. (2020)
Pavito	Conic OA	JuMP	Coey et al. (2020)
SBB (Simple Branch and Bound)	NLP-BB	GAMS, NEOS	Bussieck & Drud (2001)
SCIP (Solving Constraint Integer Programs)	Polyhedral OA and Spatial BB	MPL, C, GAMS, JuMP, MATLAB, NEOS, Java, Pyomo, Python	Gleixner et al. (2017)
SHOT (Supporting Hyperplane Optimization Toolkit)	Polyhedral OA	C++, GAMS	Lundell & Kronqvist (2022) and Kronqvist et al. (2016)

2.7 Summary

This chapter has provided background on batch processes and BWNS, highlighting their challenges. It also presented a literature review on existing techniques for short-term scheduling of multistage multiproduct batch processes and the minimization of wastewater in multipurpose multiproduct BWN. It was noted that the simultaneous integration of scheduling and wastewater minimization in multipurpose multiproduct batch processes, particularly accounting for makespan, has not been extensively studied. Moreover, no work has yet proposed a unified framework that considers the trade-offs between minimizing both makespan and water usage. Therefore, the focus of this research is on developing a mathematical formulation to simultaneously minimize freshwater consumption and production makespan in multipurpose multiproduct batch processes.

2.8 References

Abass, M., & Majozi, T. (2016). Optimization of Integrated Water and Multiregenerator Membrane Systems. *Industrial and Engineering Chemistry Research*, *55*(7), 1995–2007. https://doi.org/10.1021/acs.iecr.5b03423

Almato, M., Sanmarti, E., Espuna, A., & Puigjaner, L. (1997). Rationalizing the water use in the batch process industry. *Computers & Chemical Engineering*, *21*, S971–S976.

Bagajewicz, M. (2000). A review of recent design procedures for water networks in refineries and process plants. In *Computers and Chemical Engineering* (Vol. 24). www.elsevier.com/locate/compchemeng

Bazolana, N. C., & Majozi, T. (2017). Optimum Water Network Design for Multipurpose Batch Plants with an Electrodialysis Central Regeneration Unit. *International Symposium on Advanced Control of Industrial Processes*, *6*, 475–480. https://doi.org/978-1-5090-4397-2/17/

Belotti, P., Lee, J., Liberti, L., Margot, F., & Wächter, A. (2009). Branching and bounds tighteningtechniques for non-convex MINLP. *Optimization Methods and Software*, *24*(4–5), 597–634. https://doi.org/10.1080/10556780903087124

Bhavsar, A., Srinivasan, B., & Srinivasan, R. (2023). Freshwater minimization in multistage multiproduct batch processes through intentional waiting. *Chemical Engineering Research and Design*, *196*, 204–213. https://doi.org/10.1016/j.cherd.2023.06.039

Bonami, P., Biegler, L. T., Conn, A. R., Cornuéjols, G., Grossmann, I. E., Laird, C. D., Lee, J., Lodi, A., Margot, F., Sawaya, N., & Wächter, A. (2008). An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optimization*, *5*(2), 186–204. https://doi.org/10.1016/j.disopt.2006.10.011

Bonami, P., Kilinç, M., & Linderoth, J. (2011). Algorithms and Software for Convex Mixed Integer Nonlinear Programs.

Boukouvala, F., Misener, R., & Floudas, C. A. (2016). Global optimization advances in Mixed-Integer Nonlinear Programming, MINLP, and Constrained Derivative-Free Optimization, CDFO. In *European Journal of Operational Research* (Vol. 252, Issue 3, pp. 701–727). Elsevier B.V. https://doi.org/10.1016/j.ejor.2015.12.018

Boyd, S., & Mattingley, J. (2010). Branch and Bound Methods.

Buabeng-Baidoo, E., Mafukidze, N., Pal, J., Tiwari, S., Srinivasan, B., Majozi, T., & Srinivasan, R. (2017). Study of water reuse opportunities in a large-scale milk processing plant through process integration. *Chemical Engineering Research and Design*, *121*, 81–91. https://doi.org/10.1016/j.cherd.2017.02.031

Budak Duhbacı, T., Özel, S., & Bulkan, S. (2021). Water and energy minimization in industrial processes through mathematical programming: A literature review. In *Journal of Cleaner Production* (Vol. 284). Elsevier Ltd. https://doi.org/10.1016/j.jclepro.2020.124752

Bussieck, M. R., & Drud, A. (2001). SBB: A new solver for mixed integer nonlinear programming. *Talk*.

Byrd, R. H., Nocedal, J., & Waltz, R. A. (2006). KNITRO: An Integrated Package for Nonlinear Optimization. In *Large-Scale Nonlinear Optimization. Non-convex Optimization and Its Applications* (Vol. 83). Springer. https://doi.org/10.1007/0-387-30065-1_4

Castro, P. M., & Grossmann, I. E. (2005). New continuous-time MILP model for the short-term scheduling of multistage batch plants. *Industrial and Engineering Chemistry Research*, *44*(24), 9175–9190. https://doi.org/10.1021/ie050730l

Castro, P. M., Grossmann, I. E., & Novais, A. Q. (2006). Two new continuous-time models for the scheduling of multistage batch plants with sequence dependent changeovers. *Industrial and Engineering Chemistry Research*, *45*(18), 6210–6226. https://doi.org/10.1021/ie060449m

Castro, P. M., & Novais, A. Q. (2009). Scheduling multistage batch plants with sequence-dependent changeovers. *AIChE Journal*, *55*(8), 2122–2137. https://doi.org/10.1002/aic.11799

Cerdá, J., Henning, G. P., & Grossmann, I. E. (1997). *A Mixed-Integer Linear Programming Model for Short-Term Scheduling of Single-Stage Multiproduct Batch Plants with Parallel Lines*. https://pubs.acs.org/sharingguidelines

Chen, C. L., & Lee, J. Y. (2008). A graphical technique for the design of water-using networks in batch processes. *Chemical Engineering Science*, *63*(14), 3740–3754. https://doi.org/10.1016/j.ces.2008.04.047

Cheng, K. F., & Chang, C. T. (2007). Integrated water network designs for batch processes. *Industrial and Engineering Chemistry Research*, *46*(4), 1241–1253. https://doi.org/10.1021/ie0611150

Coey, C., Lubin, M., & Vielma, J. P. (2020). Outer approximation with conic certificates for mixed-integer convex problems. *Mathematical Programming Computation*, *12*(2), 249–293. https://doi.org/10.1007/s12532-020-00178-3

Conejo, A. J., Castillo, E., Minguez, R., & Garcia-Bertrand, R. (2006). *Decomposition techniques in mathematical programming: Engineering and science applications*. Springer.

Croft, J., Alam, M. N., Machakaire, N., Srinivasan, B., Srinivasan, R., & Majozi, T. (2021). Synthesis of an Optimal Schedule and Water Network for a Multipurpose Multiproduct Textile Industry through a Sequential MILP-NLP Technique. *Industrial and Engineering Chemistry Research*, *60*(45), 16363–16377. https://doi.org/10.1021/acs.iecr.1c02661

Duran, M. A., & Grossmann, I. E. (1986). An Outer-Approximation Algorithm for a class of Mixed-Integer Nonlinear Programs. In *Mathematical Programming* (Vol. 36).

Dutta, Suman. (2016). *Optimization in chemical engineering*. Cambridge University Press, is part of the University of Cambridge.

Edgar, T. F. ., Himmelblau, D. M. ., & Lasdon, L. S. . (2001). *Optimization of chemical processes*. McGraw-Hill.

El-Halwagi, M. M. (2017). *Sustainable Design Through Process Integration: Fundamentals and Applications to Industrial Pollution Prevention, Resource Conservation, and Profitability Enhancement*.

Erdirik-Dogan, M., & Grossmann, I. E. (2007). *A slot-based formulation for the short-term scheduling of multi-stage, multi-product batch plants with resource constraints and sequence-dependent changeovers*.

Faria, D. C., & Bagajewicz, M. J. (2010). On the Appropriate Modelling of Process Plant Water Systems. *AIChE Journal*, *56*(3), 668–689. https://doi.org/10.1002/aic.11983

Ferrer-Nadal, S., Capón-García, E., Méndez, C. A., & Puigjaner, L. (2008). Material transfer operations in batch scheduling. A critical modelling issue. *Industrial and Engineering Chemistry Research*, *47*(20), 7721–7732. https://doi.org/10.1021/ie800075u

Floudas, C. A., & Lin, X. (2004). Continuous-time versus discrete-time approaches for scheduling of chemical processes: A review. *Computers and Chemical Engineering*, *28*(11), 2109–2129. https://doi.org/10.1016/j.compchemeng.2004.05.002

Foo, D. C. Y. (2009). State-of-the-art review of pinch analysis techniques for Water network synthesis. *Industrial and Engineering Chemistry Research*, *48*(11), 5125–5159. https://doi.org/10.1021/ie801264c

Foo, D. C. Y., Manan, Z. A., & Tan, Y. L. (2005). Synthesis of maximum water recovery network for batch process systems. *Journal of Cleaner Production*, *13*(15), 1381–1394. https://doi.org/10.1016/j.jclepro.2005.04.013

Geoffrmn, A. M., & Balakrishnan, V. (1972). Generalized Benders Decomposition 1 Communicated by A. In *JOURNAL OF OPTIMIZATION THEORY AND APPLICATIONS* (Vol. 10, Issue 4).

Ghazouani, S., Zoughaib, A., & Pelloux-Prayer, S. (2015). Simultaneous heat integrated resource allocation network targeting for total annual cost considering non-isothermal mixing. *Chemical Engineering Science*, *134*, 385–398. https://doi.org/10.1016/j.ces.2015.05.027

Gleixner, A., Eifler, L., Gally, T., Gamrath, G., Gemander, P., Gottwald, R. L., Hendel, G., Hojny, C., Koch, T., Miltenberger, M., Uller, B. M. ¨, Pfetsch, M. E., Puchert, C., Rehfeldt, D., Schl¨osser, F., Schl¨osser, S., Serrano, F., Shinano, Y., Viernickel, J. M., … Witzig, J. (2017). *Zuse Institute Berlin The SCIP Optimization Suite 5.0*. http://www.zib.de

Glover, F. (1975). IMPROVED LINEAR INTEGER PROGRAMMING FORMULATIONS OF NONLINEAR INTEGER PROBLEMS. *Management Science*, *22*(4), 455–460. https://doi.org/10.1287/mnsc.22.4.455

Goode, P. (2004). *What Is Mathematical Modelling?*

Gouws, J. F., Majozi, T., & Gadalla, M. (2008). Flexible mass transfer model for water minimization in batch plants. *Chemical Engineering and Processing: Process Intensification*, *47*(12), 2323–2335. https://doi.org/10.1016/j.cep.2008.01.008

Grossmann, I. E., & Biegler, L. T. (2004). Part II. Future perspective on optimization. *Computers and Chemical Engineering*, *28*(8), 1193–1218. https://doi.org/10.1016/j.compchemeng.2003.11.006

Grunow, M., & Lehmann, M. (2002). Campaign planning for multi-stage batch processes in the chemical industry. In *OR Spectrum* (Vol. 24). Springer-Verlag.

De Souza, S. G. U., Xavier, M. F., Da Silva, A., & Ulson De Souza, A. A. (2011). Water reuse and wastewater minimization in chemical industries using differentiated regeneration of contaminants. *Industrial and Engineering Chemistry Research*, *50*(12), 7428–7436. https://doi.org/10.1021/ie200305z

Han, Y., Yan, X., & Gu, X. (2022). Novel hybrid discrete differential evolution algorithm for the multi-stage multi-purpose batch plant scheduling problem. *Applied Soft Computing*, *115*. https://doi.org/10.1016/j.asoc.2021.108262

Harjunkoski, I., & Grossmann, I. E. (2002). Decomposition techniques for multistage scheduling problems using mixed-integer and constraint programming methods. In *Computers and Chemical Engineering* (Vol. 26). www.elsevier.com/locate/compchemeng

Hunting, M. (2011). *The AIMMS Outer Approximation Algorithm for MINLP (using GMP functionality)*. https://how-to.aimms.com/Articles/366/366-aoa-for-minlp.html

Ierapetritou, M. G., & Floudas, C. A. (1998a). Effective continuous-time formulation for short-term scheduling. 1. Multipurpose batch processes. *Industrial and Engineering Chemistry Research*, *37*(11), 4341–4359. https://doi.org/10.1021/ie970927g

Ierapetritou, M. G., & Floudas, C. A. (1998b). Effective continuous-time formulation for short-term scheduling. 1. Multipurpose batch processes. *Industrial and Engineering Chemistry Research*, *37*(11), 4341–4359. https://doi.org/10.1021/ie970927g

Janak, S. L., Lin, X., & Floudas, C. A. (2004). Enhanced Continuous-Time Unit-Specific Event-Based Formulation for Short-Term Scheduling of Multipurpose Batch Processes: Resource Constraints and Mixed Storage Policies. *Industrial and Engineering Chemistry Research*, *43*(10), 2516–2533. https://doi.org/10.1021/ie0341597

Jezowski, J. (2010). Review of water network design methods with literature annotations. *Industrial and Engineering Chemistry Research*, *49*(10), 4475–4516. https://doi.org/10.1021/ie901632w

Karuppiah, R., & Grossmann, I. E. (2008). Global optimization of multiscenario mixed integer nonlinear programming models arising in the synthesis of integrated water networks under uncertainty. *Computers and Chemical Engineering*, *32*(1–2), 145–160. https://doi.org/10.1016/j.compchemeng.2007.03.007

Khor, C. S., Chachuat, B., & Shah, N. (2014). Optimization of water network synthesis for single-site and continuous processes: Milestones, challenges, and future directions. In *Industrial and Engineering Chemistry Research* (Vol. 53, Issue 25, pp. 10257–10275). American Chemical Society. https://doi.org/10.1021/ie4039482

Kim, M., Jung, J. H., & Lee, I. B. (1996). Optimal scheduling of multiproduct batch processes for various intermediate storage policies. *Industrial and Engineering Chemistry Research*, *35*(11), 4058–4066. https://doi.org/10.1021/ie9601817

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

Koop, L., Ramos, N. M. do V., Bonilla-Petriciolet, A., Corazza, M. L., & Voll, F. A. P. (2024). A Review of Stochastic Optimization Algorithms Applied in Food Engineering. *International Journal of Chemical Engineering*, *2024*, 1–31. https://doi.org/10.1155/2024/3636305

Kröger, O., Coffrin, C., Hijazi, H., & Nagarajan, H. (2018). Juniper: An Open-Source Nonlinear Branch-and-Bound Solver in Julia. In *Integration of Constraint Programming, Artificial Intelligence, and Operations Research*. Springer.

Kronqvist, J., Bernal, D. E., Lundell, A., & Grossmann, I. E. (2019). A review and comparison of solvers for convex MINLP. *Optimization and Engineering*, *20*(2), 397–455. https://doi.org/10.1007/s11081-018-9411-8

Kronqvist, J., Lundell, A., & Westerlund, T. (2016). The extended supporting hyperplane algorithm for convex mixed-integer nonlinear programming. *Journal of Global Optimization*, *64*(2), 249–272. https://doi.org/10.1007/s10898-015-0322-3

Lee, H., & Maravelias, C. T. (2018). Combining the advantages of discrete- and continuous-time scheduling models: Part 1. Framework and mathematical formulations. *Computers and Chemical Engineering*, *116*, 176–190. https://doi.org/10.1016/j.compchemeng.2017.12.003

Lee, J., & Leyffer, S. (2012). *Mixed Integer Nonlinear Programming: The IMA Volumes in Mathematics and its Applications* (Vol. 154). Springer. http://www.springer.com/series/811

Li, Z., & Majozi, T. (2019). Optimal Design of Batch Water Network with a Flexible Scheduling Framework. *Industrial and Engineering Chemistry Research*, *58*(22), 9500–9511. https://doi.org/10.1021/acs.iecr.9b00399

Lin, Y., & Schrage, L. (2009). The global solver in the LINDO API. *Optimization Methods and Software*, *24*(4–5), 657–668. https://doi.org/10.1080/10556780902753221

Liu, Y., & Karimi, I. A. (2007). Scheduling multistage, multiproduct batch plants with nonidentical parallel units and unlimited intermediate storage. *Chemical Engineering Science*, *62*(6), 1549–1566. https://doi.org/10.1016/j.ces.2006.11.053

Lundell, A., & Kronqvist, J. (2022). Polyhedral approximation strategies for non-convex mixed-integer nonlinear programming in SHOT. *Journal of Global Optimization*, *82*(4), 863–896. https://doi.org/10.1007/s10898-021-01006-1

Mahajan, A., Leyffer, S., Linderoth, J., Luedtke, J., & Munson, T. (2021). Minotaur: a mixed-integer nonlinear optimization toolkit. *Mathematical Programming Computation*, *13*(2), 301–338. https://doi.org/10.1007/s12532-020-00196-1

Maier, H. R., Razavi, S., Kapelan, Z., Matott, L. S., Kasprzyk, J., & Tolson, B. A. (2019). Introductory overview: Optimization using evolutionary algorithms and other metaheuristics. In *Environmental Modelling and Software* (Vol. 114, pp. 195–213). Elsevier Ltd. https://doi.org/10.1016/j.envsoft.2018.11.018

Majozi, T. (2005a). An effective technique for wastewater minimisation in batch processes. *Journal of Cleaner Production*, *13*(15), 1374–1380. https://doi.org/10.1016/j.jclepro.2005.04.016

Majozi, T. (2005b). Wastewater minimisation using central reusable water storage in batch plants. *Computers and Chemical Engineering*, *29*(7), 1631–1646. https://doi.org/10.1016/j.compchemeng.2005.01.003

Majozi, T. (2010). Batch Chemical Process Integration: Analysis, Synthesis and Optimization. In *Batch Chemical Process Integration: Analysis, Synthesis and Optimization*. Springer Netherlands. https://doi.org/10.1007/978-90-481-2588-3_1

Majozi, T., Brouckaert, C. J., & Buckley, C. A. (2006). A graphical technique for wastewater minimisation in batch processes. *Journal of Environmental Management*, *78*(4), 317–329. https://doi.org/10.1016/j.jenvman.2005.04.026

Majozi, T., & Gouws, J. F. (2009). A mathematical optimization approach for wastewater minimisation in multipurpose batch plants: Multiple contaminants. *Computers and Chemical Engineering*, *33*(11), 1826–1840. https://doi.org/10.1016/j.compchemeng.2009.06.008

Majozi, T., & Zhu, X. X. (2001). A novel continuous-time MILP formulation for multipurpose batch plants. 1. Short-term scheduling. *Industrial and Engineering Chemistry Research*, *40*(25), 5935–5949. https://doi.org/10.1021/ie0005452

Majozi T., Esmael R. S., & Lee J. (2017). *Understanding Batch Chemical Processes: Modelling and Case Studies*. Taylor & Francis Group.

Marchetti, P. A., Méndez, C. A., & Cerdá, J. (2012). Simultaneous lot sizing and scheduling of multistage batch processes handling multiple orders per product. *Industrial and Engineering Chemistry Research*, *51*(16), 5762–5780. https://doi.org/10.1021/ie202275y

McCormick, G. P. (1976). Computability of Global Solutions to Factorable Non-Convex Programs: Part I – Convex Underestimating Problems*. In *Mathematical Programming* (Vol. 10). North-HoUand Publishing Company.

Melo, W., Fampa, M., & Raupp, F. (2020). An overview of MINLP algorithms and their implementation in Muriqui Optimizer. *Annals of Operations Research*, *286*(1–2), 217–241. https://doi.org/10.1007/s10479-018-2872-5

Méndez, C. A., Cerdá, J., Grossmann, I. E., Harjunkoski, I., & Fahl, M. (2006). State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Computers and Chemical Engineering*, *30*(6–7), 913–946. https://doi.org/10.1016/j.compchemeng.2006.02.008

Meng, X., Liu, Y., Gao, X., & Zhang, H. (2014). LNCS 8794 - A New Bio-inspired Algorithm: Chicken Swarm Optimization. In *LNCS* (Vol. 8794).

Merchan, A. F., Lee, H., & Maravelias, C. T. (2016). Discrete-time mixed-integer programming models and solution methods for production scheduling in multistage facilities. *Computers and Chemical Engineering*, *94*, 387–410. https://doi.org/10.1016/j.compchemeng.2016.04.034

Misener, R., & Floudas, C. A. (2014). ANTIGONE: Algorithms for coNTinuous / Integer Global Optimization of Nonlinear Equations. *Journal of Global Optimization*, *59*(2–3), 503–526. https://doi.org/10.1007/s10898-014-0166-2

Modi, A. K., & Karimi, I. A. (1989). Design of Multiproduct Batch Processes with Finite Intermediate Storage. In *Computers chem. Engng* (Vol. 13, Issue 1).

Mostafaei, H., & Harjunkoski, I. (2020). Continuous-time scheduling formulation for multipurpose batch plants. *AIChE Journal*, *66*(2). https://doi.org/10.1002/aic.16804

Nema, A., Srinivasan, B., Majozi, T., & Srinivasan, R. (2021). A simple strategy to maximize water-reuse in multistage, multiproduct batch processes. *Chemical Engineering Research and Design*, *168*, 327–339. https://doi.org/10.1016/j.cherd.2021.02.012

Novara, F. M., Novas, J. M., & Henning, G. P. (2016). A novel constraint programming model for large-scale scheduling problems in multiproduct multistage batch plants: Limited resources and campaign-based operation. *Computers and Chemical Engineering*, *93*, 101–117. https://doi.org/10.1016/j.compchemeng.2016.04.030

Olesen, S. G., Polley, G. T., & Montagna, J. M. (1997). A simple methodology for the design of water networks handling single contaminants. *Chemical Engineering Research and Design*, *75*(4), 420–426. https://doi.org/10.1205/026387697523886

Pinto, J. M., & Grossmann, I. E. (1995). A Continuous Time Mixed Integer Linear Programming Model for Short Term Scheduling of Multistage Batch Plants. *Industrial and Engineering Chemistry Research*, *34*(9), 3037–3051. https://doi.org/10.1021/ie00048a015

Prasad, P., & Maravelias, C. T. (2008). Batch selection, assignment and sequencing in multi-stage multi-product processes. *Computers and Chemical Engineering*, *32*(6), 1106–1119. https://doi.org/10.1016/j.compchemeng.2007.06.012

Rangaiah, G. P. (2010). *Stochastic global optimization: Techniques and applications in chemical engineering* (Vol. 2). World Scientific Publishing.

Ryoo, H. S., & Sahinidis, N. V. (1996). A Branch-and-Reduce Approach to Global Optimization. In *Journal of Global Optimization* (Vol. 8). Kluwer Academic Publishers.

Samsatli, N. J., & Papageorgiou, L. G. (2003). A modelling framework for the synthesis and optimization of multi-product multi-stage batch plants. *Computers & Chemical Engineering*, *27*, 1761--1782.

Sanmartí, E., Friedler, F., & Puigjaner, L. (1998). Combinatorial technique for short term scheduling of multipurpose batch plants based on schedule-graph representation. *Computers and Chemical Engineering*, *22*(SUPPL.1), 847–850. https://doi.org/10.1016/s0098-1354(98)00163-x

Schilling, G., & Pantelides, C. C. (1996). A simple continuous-time process scheduling formulation and a novel solution algorithm. *Computers & Chemical Engineering*, *20*, S1221–S1226. https://doi.org/https://doi.org/10.1016/0098-1354(96)00211-6

Seid, R., & Majozi, T. (2012). A novel technique for prediction of time points for scheduling of multipurpose batch plants. *Chemical Engineering Science*, *68*(1), 54–71. https://doi.org/10.1016/j.ces.2011.08.048

Shah, N., Pantelides, C. C., & Sargent, R. W. H. (1993). A general algorithm for short-term scheduling of batch operations - II. Computational Issues. *Computers and Chemical Engineering*, *17*(2), 229–244.

Shaik, M. A., & Vooradi, R. (2017). Short-term scheduling of batch plants: Reformulation for handling material transfer at the same event. *Industrial and Engineering Chemistry Research*, *56*(39), 11175–11185. https://doi.org/10.1021/acs.iecr.7b00519

Shi, B., Qian, X., Sun, S., & Yan, L. (2017). Rule-based scheduling of multi-stage multi-product batch plants with parallel units. *Chinese Journal of Chemical Engineering*, *25*(8), 1022–1036. https://doi.org/10.1016/j.cjche.2017.03.014

Sundaramoorthy, A., & Karimi, I. A. (2005). A simpler better slot-based continuous-time formulation for short-term scheduling in multipurpose batch plants. *Chemical Engineering Science*, *60*(10), 2679–2702. https://doi.org/10.1016/j.ces.2004.12.023

Sundaramoorthy, A., & Maravelias, C. T. (2008a). Modelling of storage in batching and scheduling of multistage processes. *Industrial and Engineering Chemistry Research*, *47*(17), 6648–6660. https://doi.org/10.1021/ie701737a

Sundaramoorthy, A., & Maravelias, C. T. (2008b). Simultaneous batching and scheduling in multistage multiproduct processes. *Industrial and Engineering Chemistry Research*, *47*(5), 1546–1555. https://doi.org/10.1021/ie070944y

Susarla, N., Li, J., & Karimi, I. A. (2010). A novel approach to scheduling multipurpose batch plants using unit-slots. *AIChE Journal*, *56*(7), 1859–1879. https://doi.org/10.1002/aic.12120

Tawarmalani, M., & Sahinidis, N. V. (2005). A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming*, *103*(2), 225–249. https://doi.org/10.1007/s10107-005-0581-8

Tsai, M. J., & Chang, C. T. (2001). Water usage and treatment network design using genetic algorithms. *Industrial and Engineering Chemistry Research*, *40*(22), 4874–4888. https://doi.org/10.1021/ie010011g

Ujang Z, Wong C. L., & Manan Z. (2002). Industrial wastewater minimization using water pinch analysis: a case study on an old textile plant. *Water Science & Technology*, *46*(11–12), 77–84.

Viswanathan, J., & Grossmann, E. (1990). A Combined Penalty Function Outer-Approximation Method MINLP Optimization. In *Computers c&m. Engng* (Vol. 14, Issue 7).

Vooradi, R., & Shaik, M. A. (2013). Rigorous unit-specific event-based model for short-term scheduling of batch plants using conditional sequencing and unit-wait times. *Industrial and Engineering Chemistry Research*, *52*(36), 12950–12972. https://doi.org/10.1021/ie303294k

Wan Alwi, S. R., Ismail, A., Manan, Z. A., & Handani, Z. B. (2011). A new graphical approach for simultaneous mass and energy minimisation. *Applied Thermal Engineering*, *31*(6–7), 1021–1030. https://doi.org/10.1016/j.applthermaleng.2010.11.026

Wang, Y., & Smith, R. (1995). Time Pinch Analysis. *Chemical Engineering Research and Design*, *73*, 905–914.

Westerlund, T. (2002). Solving Pseudo-Convex Mixed Integer Optimization Problems by Cutting Plane Techniques. In *Optimization and Engineering* (Vol. 3).

Westerlund, T., & Pettersson, F. (1995). An extended cutting plane method for solving convex MINLP problems. *Computers and Chemical Engineering*, *19*(SUPPL. 1), 131–136. https://doi.org/10.1016/0098-1354(95)87027-X

Williams, P. H. (2013). *Model Building in Mathematical Programming* (5th ed.). John Wiley & Sons, Ltd.

Winston, W. L. (2004). *Operations Research: Applications and Algorithms* (A. Day & K. Brayton, Eds.; Fourth). Brooks/Cole - Thomson Learning. www.duxbury.com

Yongjian, L., Xigang, Y., & YiqingF, L. (2007). Synthesis of Water Utilization System Using Concentration Interval Analysis Method (II) Discontinuous Process. In *Chin. J. Chem. Eng* (Vol. 15, Issue 3).

Zeballos, L. J., Novas, J. M., & Henning, G. P. (2011). A CP formulation for scheduling multiproduct multistage batch plants. *Computers and Chemical Engineering*, *35*(12), 2973–2989. https://doi.org/10.1016/j.compchemeng.2011.01.043

Zhang, X., & Sargent, R. W. H. (1996). The optimal operation of mixed production facilities - A general formulation and some approaches for the solution. In *Computers them. Engng* (Vol. 20, Issue 611).

Zhou, L., Xu, K., Cheng, X., Xu, Y., & Jia, Q. (2017). Study on optimizing production scheduling for water-saving in textile dyeing industry. *Journal of Cleaner Production*, *141*, 721–727. https://doi.org/10.1016/j.jclepro.2016.09.047

Chapter 3

Model Formulation

3.1 Introduction

This chapter entails the development of an optimization model designed to address the wastewater issues in multipurpose, multiproduct plants, with a focus on balancing production efficiency and water usage. It is organized into three main sections, beginning with the construction of a superstructure that forms the foundation for the model. Based on the problem statement presented in Chapter 1, this superstructure guides the formulation of the Water Network Model, detailed in Section 3.5.2. The key assumptions underlying the model are then discussed to clarify the basis of the formulation. Finally, Section 3.5 provides a comprehensive set of mathematical constraints for both the Production Scheduling Model and the Water Network Model.

3.2 Superstructure Representation

The following superstructure serves as a blueprint for the Water Network Model, capturing the intricate interactions between water demand, supply, timing, and contaminant levels across multiple stages within a parallel-configured, multipurpose, multiproduct textile facility. This framework represents the solution space for the proposed research, as illustrated in Figure 3.1. The typical layout of a textile plant includes multiple units j/j' arranged in parallel lines with different fabric shades I. In each unit, a fabric F, is processed through various stages N, where it requires freshwater, FW, and generates wastewater, WW, at the end of each stage. In this configuration, units from lighter-shaded lines can be directly integrated with units from subsequent darker-shaded lines, either through direct reuse/recycle or indirectly through storage tanks, which are used to address timing mismatches when processes do not align perfectly. Wastewater can be reused or recycled within the same unit from which it originated,

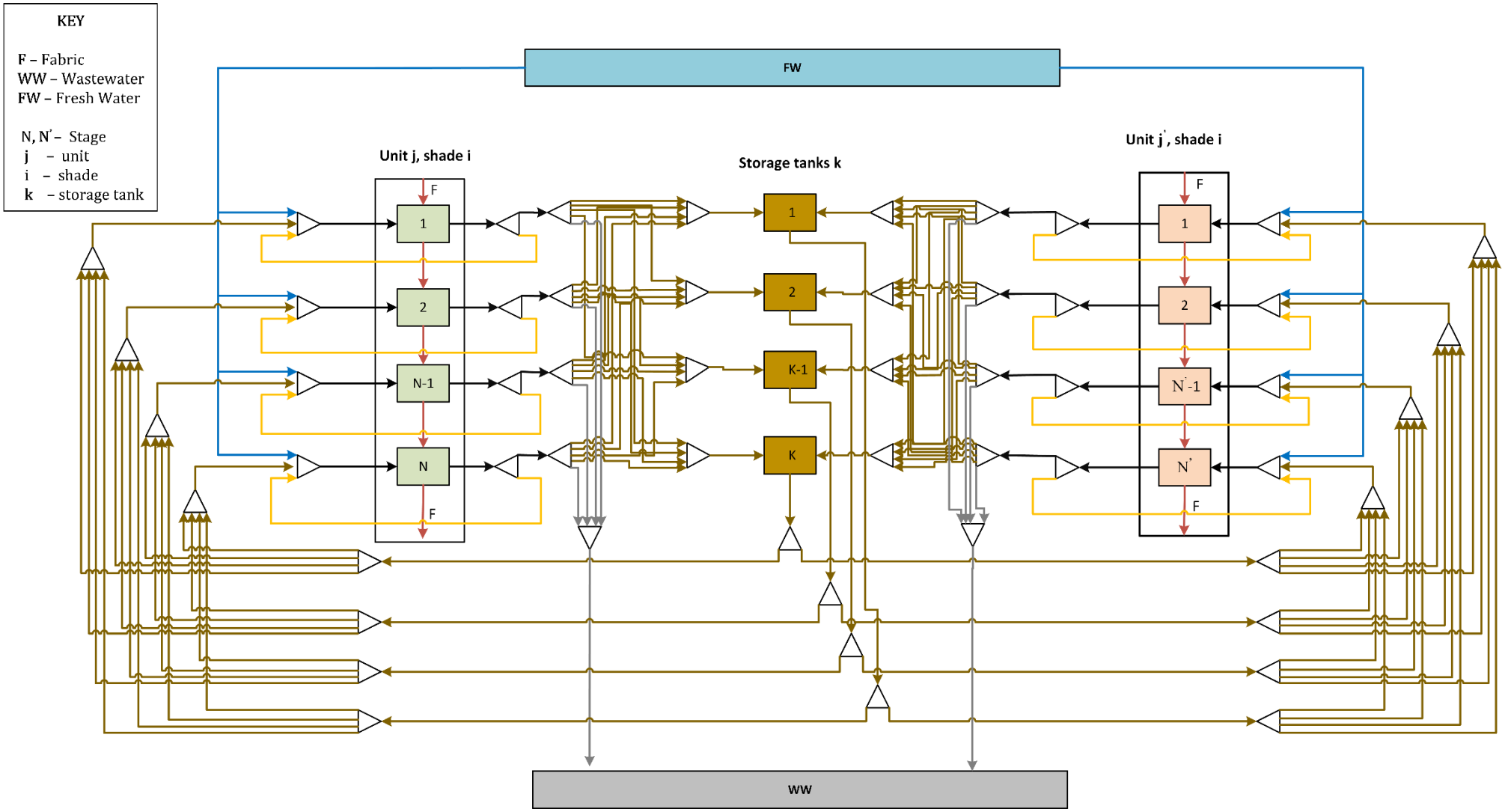


Figure 3.1. General superstructure representation of the water network

or it can be directed to preceding stages within the same line, provided it meets the threshold inlet contaminant concentration requirements of the receiving stage. If wastewater fails to meet these requirements, it is removed from the process and discharged as *WW*.

3.3 Assumptions

The proposed mathematical model is based on the following assumptions:

- Setup time is included in the task duration
- Each batch or shade is treated as a single product with uniform processing across stages, without interruptions or changes in demand during production
- Parallel lines are assumed to be perfectly synchronized in terms of process timing, i.e., batch processing starts simultaneously and continues until the required demand is met. This simplifies the model by avoiding the need to account for timing mismatches, delays, and inter-line resource conflicts in the production schedule
- No washing operations are required after completion of a task
- Consecutive batches processed in the same unit do not interfere with each other, so contaminants from the previous batch are not carried over
- Freshwater is assumed to be free of contaminants.

3.4 Nomenclature

A list of the sets, parameters and variables used in the developed model are presented below:

Sets

J $\{j|j \ represents \ a \ processing \ unit\}$

N $\{n|n \ represents \ a \ stage\}$

P $\{p|p \ represents \ time \ point\}$

K $\{k|k \ represents \ a \ water \ storage \ tank\}$

S_{in} $\{s_{in}|s_{in} \ represents \ a \ shade \ (white, light, medium, dark)\}$

$S_{in,j}$ $\{s_{in,j}, s_{in,j'}|s_{in,j}, s_{in,j'} \ represents \ a \ task \ (shade \ being \ processed) \ in \ a \ unit \ j\}$

Parameters

$V_{S_{in,j}}^{U}$	Maximum capacity of unit j
$V_{S_{in,j}}^{L}$	Minimum capacity of unit j
$Q_s^0(s)$	Initial amount of state s stored
$Q^U(s)$	Maximum storage available for a state
$D(S_{out,j})$	Required number of batches for shade S_{in}
$\tau(n, s_{in,j})$	Processing duration of effective state $s_{in,j}$
$C_{in}^{U}(n, s_{in,j})$	Maximum concentration allowable for a task $s_{in,j}$
$C_{out}(n, s_{in,j})$	Discharge concentration from a task
$W^U(n, s_{in,j})$	Water quantity requirement to process a task
$Q_{ws}^{U}(k)$	Maximum wastewater storage capacity

Binary Variables

$y(n, s_{in,j}, p)$	Binary variable indicating whether a task is active
$ys_{in}(n, s_{in,j}, p)$	Binary variable showing the transfer of water from a completed task to storage
$ys_{out}(n, s_{in,j}, p)$	Binary variable showing the transfer of water from storage to unit j
$yw_r(s_{in,j}, s_{in,j'}, p)$	Binary variable associated with the existence of direct reuse from unit j to unit j'

Continuous Variables

$t_{in}(n, s_{in,j}, p)$	Starting time of a task $s_{in,j}$
$t_{out}(n, s_{in,j}, p)$	Ending time of a task $s_{in,j}$
$mw_f(n, s_{in,j}, p)$	Quantity of freshwater into stage n at event point p

$mw_{in}(n, s_{in,j}, p)$	Quantity of water into stage n at event point p
$mw_{out}(n, s_{in,j}, p)$	Quantity of water produced from stage n at event point p
$mw_e(n, s_{in,j}, p)$	Amount of effluent from unit j at event point p
$mw_r(s_{in,j}, s_{in,j'}, p)$	Amount of water directly reused at event point p
$ms_{in}(n, k, s_{in,j}, p)$	Amount of water transferred from stage n to storage k at event point p
$ms_{out}(k, n, s_{in,j}, p)$	Amount of water transferred from storage k to stage n at event point p
$q_{ws}(k, p)$	Quantity of water stored in the storage tank *k* at event point *p*
$C_{in}(n, s_{in,j}, p)$	Contaminant concentration in the water entering a task
$C_{out}(n, s_{in,j'}, p)$	Contaminant concentration from another unit j' at event point p
$CS_{out}(k, p)$	Contaminant concentration in storage tank *k* at event point *p*
H_j	Time horizon of a unit j
ϑ_{Slack}	Time penalty
MS	Makespan of the production cycle

3.5 Mathematical Model

Figure 3.2 below illustrates the sub-models that make up the overall structure of the proposed model framework.

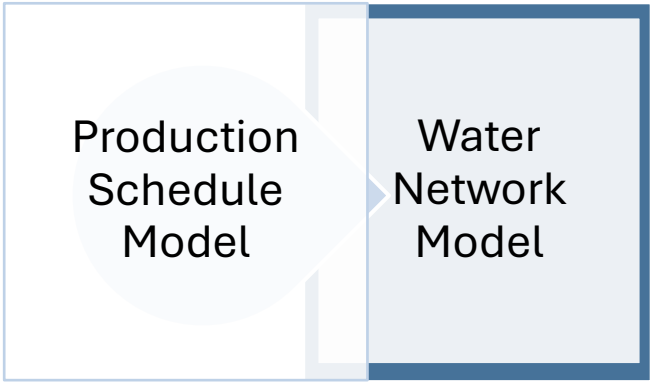


Figure 3.2. Components of the proposed formulation

3.5.1 Production Schedule Model

The production scheduling model considers the sequencing and timing of each task. Effective scheduling is crucial for optimal unit allocation, especially in multipurpose plants with shared equipment and diverse production tasks (Maravelias & Grossmann, 2004). The proposed model adopts the unit-specific event-based formulation used by Croft et al. (2021), where each event point is unique to individual units. This approach provides flexibility by allowing for unit-specific timings, which is particularly valuable for managing the unique demands and overlapping production stages common in multiproduct environments (Castro et al., 2001).

3.5.1.1 Assignment Constraints

Constraints (3.1) and (3.2) ensure that all stages occur in accordance with the recipe sequence and that stages are not processed at the same time in one unit. These constraints enforce adherence to the prescribed recipe sequence, ensuring that each stage in the production process follows the specified order without deviation. By maintaining this order, the model ensures that each intermediate stage meets the necessary quality and specification standards before advancing to the subsequent stage. They also address the exclusivity of operations within a single unit by preventing multiple stages from being processed simultaneously in the same unit.

$$\sum_{p \in P} y(n, s_{in,j}, p) = \sum_{n' \in N} \sum_{p \in P} y(n', s_{in,j}, p), \qquad \forall s_{in,j} \in S_{in,j} \tag{3.1}$$

$$y(n+1, s_{in,j}, p+1) = y(n, s_{in,j}, p), , \qquad \forall p \in P,\ n \in N,\ s_{in,j} \in S_{in,j} \tag{3.2}$$

Constraint (3.3) states that a maximum of one task can begin in a unit j at any given event p. This constraint ensures that each unit can handle only one stage at a time and avoids situations in which a unit performs multiple different tasks at one event point. The constraint is written for every stage *n*, in unit j at every event p.

$$\sum_{n \in N} \sum_{s_{in,j} \in S_{in,j}} y(n, s_{in,j}, p) \leq 1, \qquad \forall p \in P \tag{3.3}$$

3.5.1.2 Duration Constraint

The duration constraint regulates the processing time of each stage by ensuring that stages are finished after the stipulated mean processing time. In constraint (3.4), a state is produced after it has been used for the duration of the stages mean processing time. The mean processing times vary depending on the processing stages.

$$t_{out}(n, s_{in,j}, p) = t_{in}(n, s_{in,j}, p) + \tau(n, s_{in,j})\, y(n, s_{in,j}, p), \qquad \forall p \in P, \tag{3.4}$$
$$n \in N, s_{in,j} \in S_{in,j}$$

3.5.1.3 Sequence Constraints

Constraints (3.5) and (3.6) ensure that a state is produced from a processing stage before it can be used in the next stage at the next event point. Constraint (3.5) enforces a sequential relationship between successive stages within the same batch, meaning that it mandates that each stage within a unit must begin only after the preceding stage has been fully completed.

$$t_{in}(n+1, s_{in,j}, p+1) \geq t_{out}(n, s_{in,j}, p), \qquad \forall p \in P, n \in N, s_{in,j} \in S_{in,j} \tag{3.5}$$

$$t_{in}(n_1, s_{in,j}, p+1) \geq t_{out}(n, s'_{\mathrm{in,j}}, p), \qquad \forall p \in P, n, n_1 \in N, \tag{3.6}$$
$$s_{in,j} s_{in,j}{}' \in S_{in,j}$$

Constraint (3.6) extends this temporal sequencing to account for interactions between different batches processed within the same unit. Specifically, it requires that the first stage in the production sequence of a new batch starts after the final stage of the preceding batch has been completed.

3.5.1.4 Time Horizon Constraints

Constraints (3.7) and (3.8) ensure that the starting and finishing times of the tasks are accommodated within the time horizon specific to a unit. Constraint (3.7) establishes that the start time of each task within a unit must fall within the allowable time frame for the operation of that unit ensuring that no task begins prematurely, outside the designated scheduling window. Similarly, constraint (3.8) enforces the requirement that each task must be finished within the operational time horizon of that unit. By ensuring that each task concludes within the permissible time window, this constraint maintains the integrity of the overall schedule for the unit, preventing delays that could disrupt subsequent tasks and ensuring smooth transitions between tasks within the unit. This alignment is essential to minimizing idle times and avoiding task overlaps, both of which are critical to maximizing resource utilization and operational efficiency (Liu & Karimi, 2007b).

$$t_{in}(n, s_{in,j}, p) \leq H_j, \qquad \forall s_{in,j} \in S_{in,j}, p \in P, n \in N \tag{3.7}$$

$$t_{out}(n, s_{in,j}, p) \leq H_j, \qquad \forall s_{in,j} \in S_{in,j}, p \in P, n \in N \tag{3.8}$$

Constraint (3.9) ensures that the cumulative durations of all tasks assigned to a unit should be within the time horizon of interest. It states that the sum of the duration of all tasks occurring in a unit should not exceed the timespan for that unit. If the cumulative duration of scheduled tasks were to exceed the time horizon, it would not only create scheduling conflicts but it could also cause delays in downstream processes and result in a higher makespan. This constraint, therefore, sets an upper limit on the combined duration of tasks, ensuring that each unit adheres to its specified time horizon.

$$\sum_{p \in P} \sum_{s_{in,j} \in S_{in,j}} \tau(n, s_{in,j}) y(n, s_{in,j}, p) \leq H_j, \qquad \forall j \in J, n \in N \tag{3.9}$$

3.5.1.5 Demand Constraint

Constraint (3.10) represents the production requirement ensuring that the output meets or exceeds market demand. This constraint aligns the production targets with specific demand levels for each product shade, effectively linking the internal scheduling decisions of the model to external market requirements.

$$\sum_{s_{in,j} \in S_{in,j}} \sum_{p \in P} y(n_1, s_{in,j}, p) \geq D\left(S_{out,j}\right), \qquad \forall s_{in,j} \in S_{in,j}, s_{out,j} \in S_{out,j}, n \in N \tag{3.10}$$

3.5.1.6 Objective Function

The makespan for the entire production campaign would have the maximum duration. The makespan represents the total time required to complete a set of tasks (Pinedo, 2016). Constraint (3.11) defines the time horizon specific to a particular unit. It states that the sum of the duration of tasks in a unit is equal to the timespan of that particular unit. By setting the makespan to the maximum duration across all tasks in a unit, constraint (3.11) ensures that the overall production campaign is contained within a manageable timeframe. This constraint thereby guarantees that the time allocation of each unit is both comprehensive enough to accommodate the necessary tasks and limited enough to meet broader production deadlines.

$$H_j = \sum_{s^*_{in,j} \in S_{in,j}} \tau\left(n, s_{in,j}\right), \qquad \forall j \in J, n \in N \tag{3.11}$$

The objective is to minimize the time it takes to meet the required production demand. Constraint (3.12) defines the makespan as the longest timespan given by the units processing the tasks. Essentially the length of the campaign is defined by the total duration of successive batches from the unit that completes processing last.

$$MS = \max_j\left(H_j\right) \tag{3.12}$$

The objective function then becomes minimization of the makespan in constraint (3.12).

$$Min\ Z_1 = MS \tag{3.13}$$

3.5.2 Water Network Model

The Water Network Model focuses on reducing freshwater usage by identifying and maximizing opportunities for water recycle and reuse throughout various stages of production (Chew et al., 2008). This model works alongside the production schedule to ensure that production targets are met while minimizing freshwater consumption. By integrating the Water Network Model with the Production Scheduling Model, a holistic framework is created that considers both time-based and water-based constraints, simultaneously (Li & Majozi, 2019). This synergy optimizes production schedules and improves resource efficiency, providing a balanced solution that meets both operational and environmental goals for sustainability purposes. The resulting integrated model offers the dual benefits of reducing both production time (makespan) and water usage.

3.5.2.1 Unit Water Balance

Unit Water Balances track the flow of water within each process unit by ensuring that the amount of water entering, retained, leaving, and being recycled within each unit is accurately accounted for.

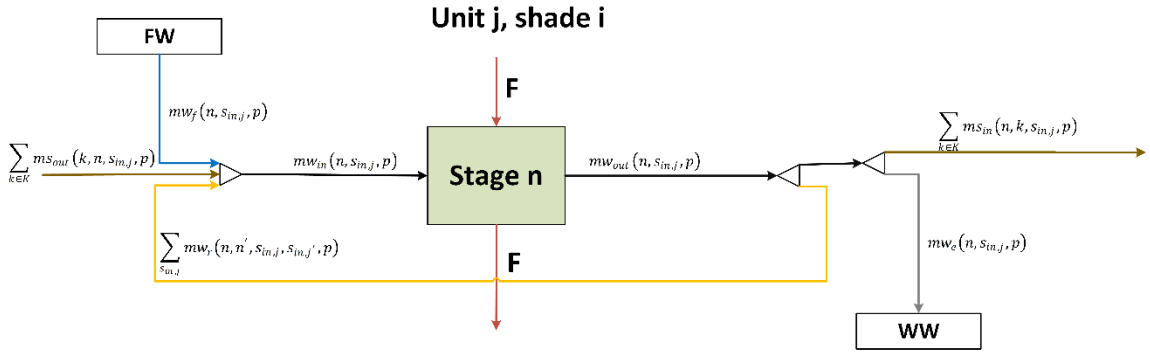


Figure 3.3. Water Network around unit j, shade i

Constraint (3.14) determines the amount of water entering a unit, specifying that the water supplied to a task within any given unit at event point *p* can be composed of both freshwater and recycled water. This recycled water may originate from storage tanks or be sourced directly from other tasks where water has already been used.

$$mw_{in}(n, s_{in,j}, p) = mw_f(n, s_{in,j}, p) + \sum_{s_{in,j}} mw_r(n, n', s_{in,j}, s_{in,j'}, p) + \sum_{k \in K} ms_{out}(k, n, s_{in,j}, p), \quad \forall s_{in,j'}, s_{in,j} \in S_{in,j}, p \in P, n \in N \tag{3.14}$$

Similarly, constraint (3.15) determines the amount of water leaving a unit, specifying that the outlet stream from a task can be managed in three primary ways: it can be disposed of as wastewater, immediately recycled by other tasks within the system, or directed to storage tanks for future use.

$$mw_{out}(n, s_{in,j}, p) = \sum_{s_{in,j}} mw_r(n, n', s_{in,j}, s_{in,j'}, p) + \sum_{k \in K} ms_{in}(n, k, s_{in,j}, p) + mw_e(n, s_{in,j}, p), \quad \forall s_{in,j'}, s_{in,j} \in S_{in,j}, p \in P, n \in N \tag{3.15}$$

Constraint (3.16) ensures that the water used in a task is conserved and reinforces a balanced, closed-loop water system using constraints (3.14) and (3.15), which respectively control the sources of incoming water and the destinations of outgoing water. It states that the amount of water entering a unit before a task commences is equal to the water leaving a unit when the task is done.

$$mw_{in}(n, s_{in,j}, p-1) = mw_{out}(n, s_{in,j}, p), \quad \forall s_{in,j} \in S_{in,j}, p \in P, n \in N \tag{3.16}$$

Constraint (3.17) limits the total volume of water entering unit *j*, ensuring it does not surpass the maximum water capacity permissible for operations in that unit. Similarly, constraint (3.18) restricts the amount of recycled water entering unit *j* to the maximum allowable water for operation in unit *j*.

$$mw_{in}(n, s_{in,j}, p) \leq W^U(n, s_{in,j}) y(n, s_{in,j}, p), \quad \forall s_{in,j} \in S_{in,j}, p \in P, n \in N \tag{3.17}$$

$$mw_r(s_{in,j}, s_{in,j'}, p) \leq W^U(n, s_{in,j}) yw_r(s_{in,j}, s_{in,j'}, p), \qquad \forall\, n \in N, \qquad (3.18)$$
$$p \in P,\ s_{in,j}, s_{in,j'} \in S_{in,j},$$

3.5.2.2 Unit Concentration Balance

The resultant concentration in the inlet of each stage is calculated by constraint (3.19). It states that the inlet contaminant concentration of a stage is made up of the contaminant concentration coming from other units as well as from storage tanks at any given event point p.

$$C_{in}(n, s_{in,j}, p) mw_{in}(n, s_{in,j}, p) \qquad (3.19)$$
$$= \sum_{s_{in,j'}} C_{out}(n, s_{in,j'}, p)\ mw_r(n, n', s_{in,j}, s_{in,j'}, p)$$
$$+ \sum_{k \in K} CS_{out}(k, p)\, ms_{out}(k, n, s_{in,j}, p), \qquad \forall s_{in,j'}, s_{in,j} \in S_{in,j},$$
$$p \in P, j \in J, n \in N$$

Constraint (3.20) ensures that the inlet contaminant concentration into a stage does not exceed the maximum allowable limit of that stage.

$$C_{in}(n, s_{in,j}, p) \leq C_{in}^U(n, s_{in,j})\, y(n, s_{in,j}, p), \qquad \forall\, s_{in,j} \in S_{in,j}, \qquad (3.20)$$
$$n \in N, j \in J, p \in P$$

3.5.2.3 Storage Water Balance

Storage tanks play a crucial role in WNS by temporarily holding water that is either surplus or insufficient for immediate use. They are especially useful when water usage does not align with the timing of production activities. Storage tanks facilitate the buffering of water flows, enabling the reuse of water that would otherwise be discarded as effluent (Chew et al., 2008).

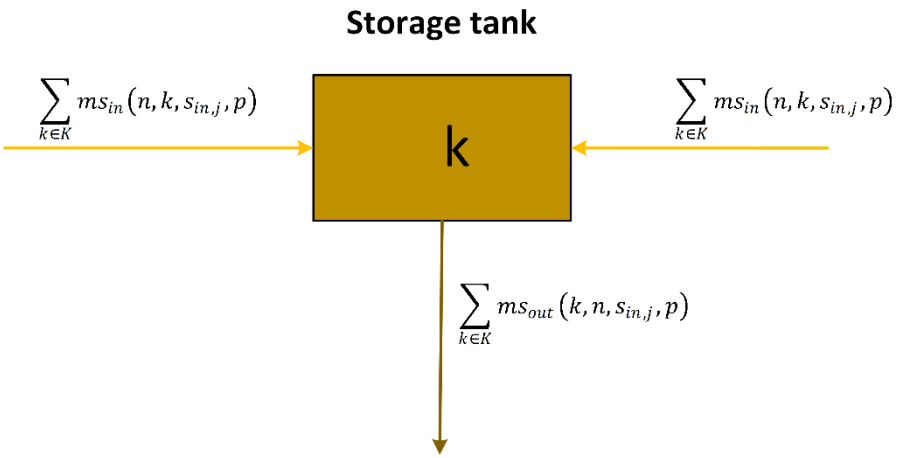


Figure 3.4. Water network around the storage tanks

Constraint (3.21) traces the water stored in each storage vessel. It states that the amount of water stored in a storage tank at any event point p is composed of the amount of water that was initially present in the tank as well as the difference between the amount of water that was transferred to and from the tank.

$$q_{ws}(k,p) = q_{ws}(k,p-1) + \sum_{s_{in,j}\in S_{in,j}} [ms_{in}(n,k,s_{in,j},p) - ms_{out}(k,n,s_{in,j},p)], \quad \forall n \in N, k \in K, p \in P \tag{3.21}$$

3.5.2.4 Storage Concentration Constraints

Constraint (3.22) determines the resulting concentration of contaminants in tank k at event point p.

$$CS_{out}(k,p)\left[q_{ws}(k,p-1) + \sum_{s_{in,j}\in S_{in,j}} ms_{in}(n,k,s_{in,j},p)\right] = \left[q_{ws}(k,p-1)CS_{out}(k,p-1) + \sum_{s_{in,j}\in S_{in,j}} ms_{in}(n,k,s_{in,j},p)\, C_{out}(n,s_{in,j})\right], \quad \forall n \in N, k \in K, p \in P \tag{3.22}$$

3.5.2.5 Storage Capacity Constraints

Constraint (3.23) ensures that the volume of water stored in the tank remains within its maximum storage capacity.

$$Q_{ws}^{U}(k) \geq q_{ws}(k,p) + \sum_{s_{in,j} \in S_{in,j}} ms_{in}\left(n,k,s_{in,j},p\right), \qquad \forall k \in K, p \in P \tag{3.23}$$

3.5.2.6 Time Horizon Constraints

Constraints (3.24) and (3.25) modify the scheduling time frame to account for timing deviations when water is directly reused or recycled between unsynchronized stages and units. Similar to constraints (3.7) and (3.8), these constraints ensure that the start and end times of all water-using tasks fit within the extended time horizon specific to each unit.

$$t_{in}\left(n,s_{in,j},p\right) \leq H_j + \vartheta_{Slack}, \qquad \forall s_{in,j} \in S_{in,j}, p \in P, n \in N \tag{3.24}$$

$$t_{out}\left(n,s_{in,j},p\right) \leq H_j + \vartheta_{Slack}, \qquad \forall s_{in,j} \in S_{in,j}, p \in P, n \in N \tag{3.25}$$

A slack variable, ϑ_{Slack}, is included in these constraints to allow for the necessary extension of the time horizon, accommodating the direct reuse and recycling processes. Particularly, this is important in scenarios where tasks may require water when it is unavailable due to timing mismatches. This variable, therefore, allows for the necessary flexibility in the scheduling framework. Consequently, the makespan is adjusted to reflect these deviations from the original time horizon. By incorporating the slack variable, the model dynamically accounts for the additional time required to accommodate water reuse and recycling operations.

$$MS = \max_{j}\left(H_j\ \vartheta_{Slack}\right) \tag{3.26}$$

Constraint (3.26) defines the new makespan as the time taken by the unit that completes processing last. This adjustment ensures that the makespan accurately represents the total duration of the production campaign, including any delays introduced by water reuse and recycling.

3.5.2.7 Objective Function

The objective function aims to minimize the net freshwater consumption, wastewater generation, and overall production time. It incorporates the slack variable, ϑ_{Slack}, introduced in the timing constraints to penalize the model when the scheduling time horizon is extended. This penalty encourages the optimization process to identify a solution where water usage is minimized while also ensuring that the time extension remains within a reasonable limit.

$$Min[Z_2] = \sum_{s_{in,j} \in S_{in,j}} \sum_{p \in P} [mw_f\,(n, s_{in,j}, p\,) + mw_e(n, s_{in,j}, p\,)] + \vartheta_{Slack}, \qquad \forall n \in N \tag{3.27}$$

This trade-off is essential to ensure that the primary objective, i.e., reducing water consumption, is achieved without unduly compromising production time, which could impact operational throughput and profitability. The optimization process must, therefore, strike a balance between resource conservation and maintaining an efficient production schedule to maximize both environmental and economic outcomes.

3.6 References

Castro, P., Barbosa-Póvoa, A. P. F. D., & Matos, H. (2001). An improved RTN continuous-time formulation for the short-term scheduling of multipurpose batch plants. *Industrial and Engineering Chemistry Research*, *40*(9), 2059–2068. https://doi.org/10.1021/ie000683r

Chew, I. M. L., Tan, R., Ng, D. K. S., Foo, D. C. Y., Majozi, T., & Gouws, J. (2008). Synthesis of direct and indirect interplant water network. *Industrial and Engineering Chemistry Research*, *47*(23), 9485–9496. https://doi.org/10.1021/ie800072r

Croft, J., Alam, M. N., Machakaire, N., Srinivasan, B., Srinivasan, R., & Majozi, T. (2021). Synthesis of an Optimal Schedule and Water Network for a Multipurpose Multiproduct Textile Industry through a Sequential MILP-NLP Technique. *Industrial and Engineering Chemistry Research*, *60*(45), 16363–16377. https://doi.org/10.1021/acs.iecr.1c02661

Li, Z., & Majozi, T. (2019). Optimal Design of Batch Water Network with a Flexible Scheduling Framework. *Industrial and Engineering Chemistry Research*, *58*(22), 9500–9511. https://doi.org/10.1021/acs.iecr.9b00399

Liu, Y., & Karimi, I. A. (2007). Scheduling multistage, multiproduct batch plants with nonidentical parallel units and unlimited intermediate storage. *Chemical Engineering Science*, *62*(6), 1549–1566. https://doi.org/10.1016/j.ces.2006.11.053

Maravelias, C. T., & Grossmann, I. E. (2004). A hybrid MILP/CP decomposition approach for the continuous time scheduling of multipurpose batch plants. *Computers and Chemical Engineering*, *28*(10), 1921–1949. https://doi.org/10.1016/j.compchemeng.2004.03.016

Pinedo, M. L. (2016). *Scheduling: Theory, algorithms, and systems* (5th ed.). Springer.

Chapter 4

Industrial Case Study

4.1 Introduction

This chapter demonstrates the applicability of the formulations developed in the previous chapter using a case study adapted from literature. The resultant production scheduling and water network models were implemented in GAMS 38.2.1 and solved using CPLEX and CONOPT3, respectively. The computer used to solve the models had the following specifications: Intel® Core™ i7-10700 CPU @ 2.90GHz, Windows 11 Education, 16 GB of RAM, and 64-bit Operating System.

4.2 Case Study

The case study used in this chapter is adapted from Nema et al. (2021), which focuses on a multipurpose multiproduct textile plant equipped with multiple dyeing units. The plant under consideration consists of six soft-flow dyeing machines, each with identical processing capacity allowing for flexible production across different products. The case study examines

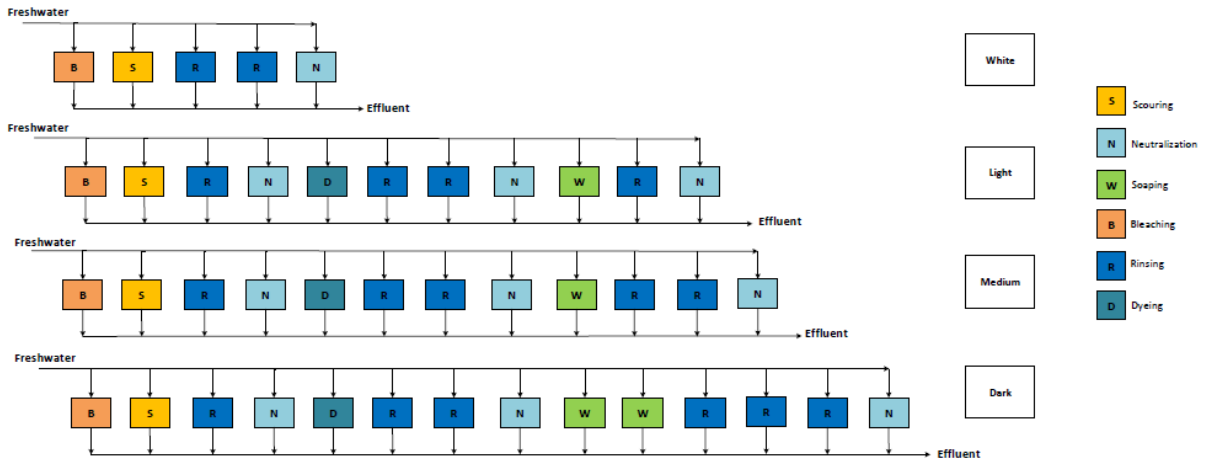


Figure 4.1. Production recipe

four distinct product shades, i.e., white, light, medium, and dark, each with its own specific set of processing stages and a unique dyeing recipe, as illustrated in Figure 4.1.

Table 4.1 presents the stage-by-stage production data for each product shade in the textile dyeing process, detailing the threshold concentrations and durations of each stage. The setup time for switching between different product shades usually varies based on whether a lighter or darker shade follows (Zhou et al., 2020). Transitioning from a light to a dark shade requires a shorter setup time, whereas transitioning from a dark shade to a light shade necessitates a longer setup time. In this case, the setup times are included in the stage durations. The concentrations of the chemicals injected in each stage have been normalized to a range of 0-1, with a value of one representing the regulatory limit. A notable characteristic of this process is the gradual decrease of effluent contaminant concentration as production moves through each stage. This trend reflects the unique configuration of textile dyeing batch operations, where certain dyeing and rinsing stages inherently dilute contaminants over time, thereby lowering concentrations at successive stages. The set demand for this case study is six batches of each product shade.

Table 4.1. Production recipe for each stage

Stage	**Inlet threshold concentration**	**Concentration of effluent**	**Duration (min)**			
			White	*Light*	*Medium*	*Dark*
1	0.93	0.94	53	28	32	32
2	0.91	0.92	123	88	96	88
3	0.89	0.90	21	18	18	18
4	0.87	0.88	16	30	30	30
5	0.84	0.86	42	202	207	234
6	0.80	0.83	-	24	19	24
7	0.76	0.79	-	6	19	24
8	0.70	0.75	-	26	26	26
9	0.65	0.69	-	22	48	48
10	0.56	0.63	-	16	19	14
11	0.46	0.55	-	29	16	19
12	0.33	0.45	-	-	39	16
13	0.19	0.32	-	-	-	16
14	0.0	0.18	-	-	-	39
Total			255	499	569	628

Each stage has a specific water requirement of 2 975 kg, which is essential for ensuring thorough dye penetration into the fabric. At the end of each stage, an equivalent amount of water, 2 975 kg, is discharged as wastewater. In the absence of water reuse, the complete product campaign requires 749 700 kg of freshwater.

4.3 Results and Discussion

The MILP scheduling problem was solved using CPLEX, which applies the branch-and-bound algorithm. The optimized schedule, depicted as a Gantt chart in Figure 4.2, shows the assignment of batches to units and the sequencing of batches in each unit. The optimal makespan achieved is 1 951 minutes. In high product variety environments like the textile industry, production scheduling is particularly challenging due to the numerous batches and varying production recipes (Grunow & Lehmann, 2002). These factors increase the dimensionality of the model, affecting the number of constraints and variables, and thus extending computational time. The unit-specific event-based formulation used assigned unique event points i.e., stages, to each unit, which effectively reduced redundant constraint overlaps between units.

The scheduling model is large-scale, with over 10 000 variables, necessitating significant computational power. The CPLEX branch-and-bound algorithm, effectively manages this large variable set, enabling the model to converge on a feasible schedule within the desired time frame. However, the complexity of the problem escalates when integrating the water network into the production schedule. While the scheduling model initially focuses on variables related to the sequencing and timing of production tasks, the inclusion of the water network introduces additional variables, such as dye concentrations, the volume of water reused per stage, and the management of storage tanks. The interactions between these variables make it challenging to predict their combined effects on the makespan and water usage. Given these complexities, two scenarios derived from the simplified case study were applied to the water network model and investigated:

- Scenario A: Two units producing two batches of two different shades (W & L)
- Scenario B: Four units producing three batches of three different shades (W, L & M)

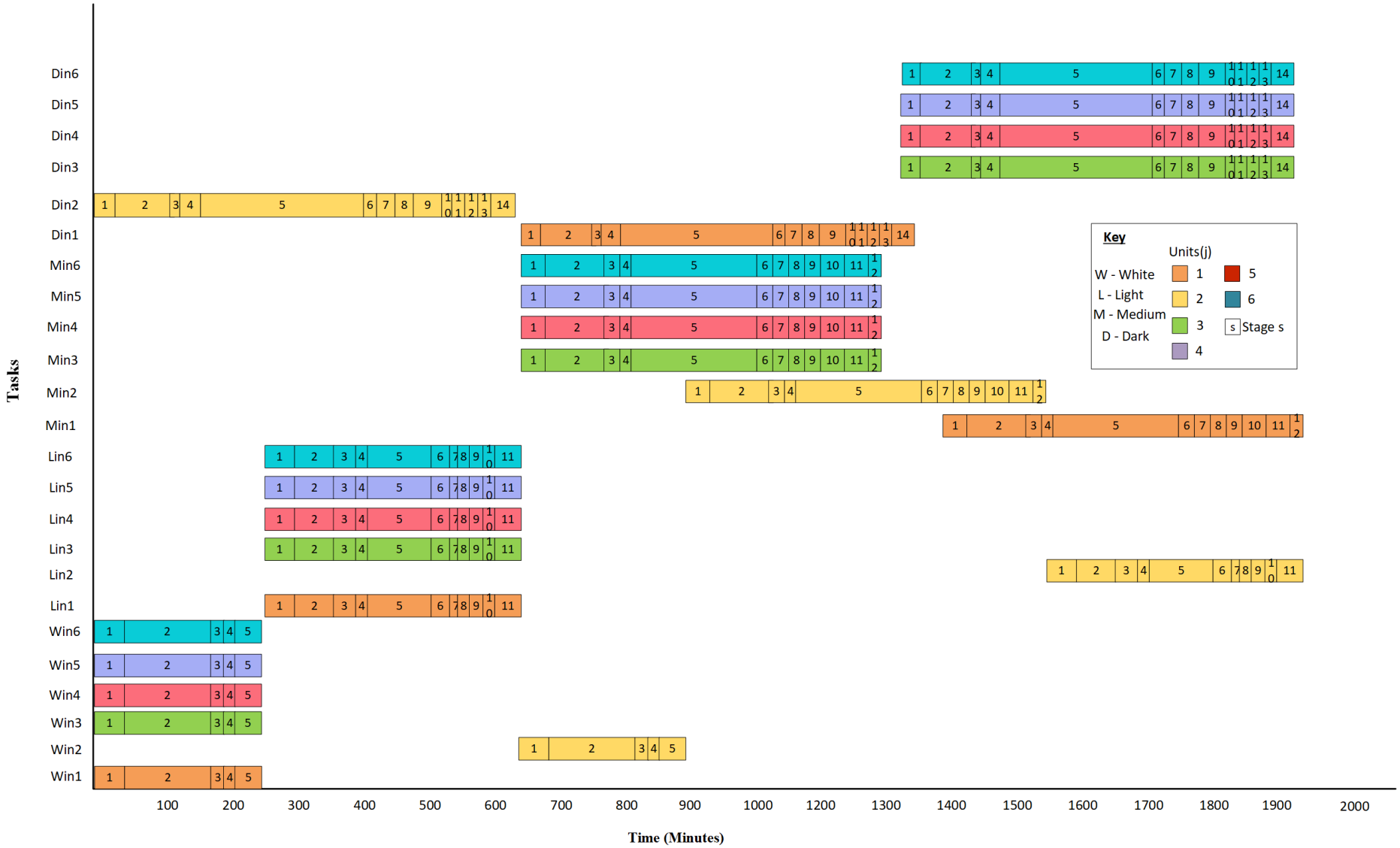


Figure 4.2. Optimal Production Schedule before water integration

The resultant MINLP problem was solved using DICOPT, a decomposition-based solver that effectively integrates CPLEX and CONOPT3 to handle the MILP and NLP aspects of the model, respectively. CONOPT3 focuses on solving the NLP constraints, particularly addressing the nonlinearity introduced by integrating the water network into the scheduling model. As a result of the additional variables introduced by this integration, the complexity of the problem grows exponentially, as reflected in the increased computational time and the larger number of constraints and variables outlined in Table 4.2. CONOPT3, as a robust NLP solver, manages this complexity by employing iterative optimization techniques to navigate the nonlinear solution space. The solver relies on algorithms such as gradient-based methods which ensure that the model converges to a viable solution despite the high-dimensional and nonlinear nature of the problem. Furthermore, it was necessary to provide upper bounds for key variables such as the makespan to prevent it from reaching infinity. Without these bounds, the slack variable can theoretically expand indefinitely, leading to impractically large solutions (Floudas & Pardalos, 2009). The bounds also help control computational complexity, as unbounded variables can lead to longer solution times and convergence issues (Grossmann, 2012).

Table 4.2. Model statistics

	Production Schedule	**Scenario A**	**Scenario B**
No. single equations	48 598	79 093	960 036
No. single variables	38 809	39 230	213 087
No. binary variables	10 584	17 239	95 148
Non-linear terms	0	486 398	9 234 336
CPU time (s)	95.92	253.78	7 135.17

The inclusion or exclusion of storage tanks are the limiting cases in wastewater minimization, with the inclusion of storage tanks enabling greater water savings and their absence resulting in minimal savings. In WNS studies, direct and indirect reuse scenarios are often compared to determine the optimal balance between network complexity, cost, and water efficiency, particularly in high water-usage industries such as textiles, chemicals, and food processing (Buabeng-Baidoo et al., 2017b). Regardless of whether storage tanks are included, the ultimate

goal is to optimize the water network by balancing water savings with operational efficiencies i.e., minimum production time.

Figures 4.3 and 4.4 illustrate the water schedules for Scenario A. In the configuration without storage tanks, water recycling is entirely dependent on the immediate availability of used water from upstream stages. In this scenario, water savings are significantly constrained, as any excess water that cannot be utilized immediately is discharged as effluent. This dependency results in a greater reliance on freshwater supplementation across various stages, as evident in the light shades of both units. Additionally, wastewater is released as effluent when no downstream processes can effectively use it, either due to the lack of suitable operations to consume the water or because the quality of the outlet water does not meet the required inlet concentration thresholds for reuse.

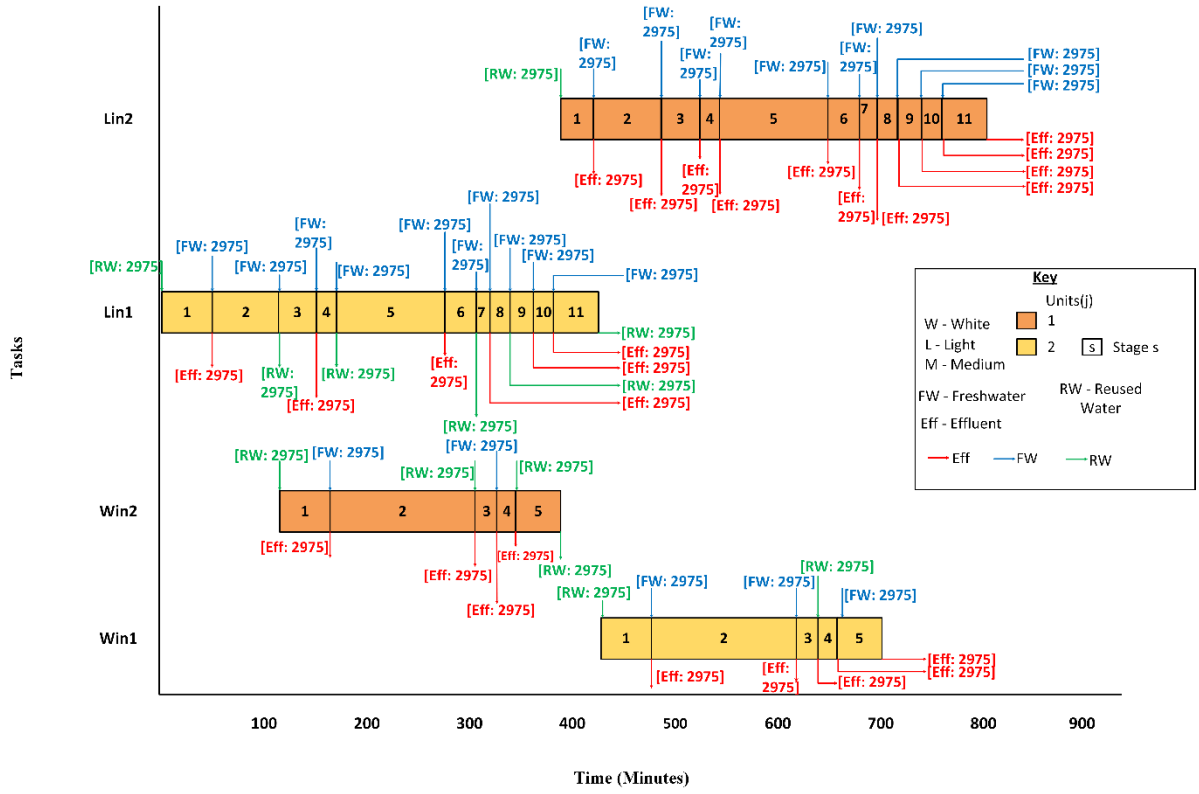


Figure 4.3. Gantt chart for Scenario A without storage tanks

The incorporation of storage tanks, as depicted in Figure 4.4, significantly reduces dependence on the immediate availability of used water, thereby enabling more effective recycling (Bhavsar et al., 2023). The Gantt chart in the figure visually represents the process schedule, with rectangular boxes denoting the operational stages across units and the storage tanks. In

this scenario, water for the light and white shades in the two units is primarily drawn from storage tanks, supplemented with freshwater only when necessary. This demonstrates the critical role of storage tanks in minimizing freshwater consumption by providing stored water for reuse and ensuring the dilution of highly concentrated streams to meet the inlet thresholds of receiving stages. The optimized solution involved the use of two shared storage tanks, each with a capacity of 5 000 kg. Additionally, the shared tanks facilitate the mixing of different water streams, effectively diluting highly contaminated streams and enhancing the feasibility of reuse.

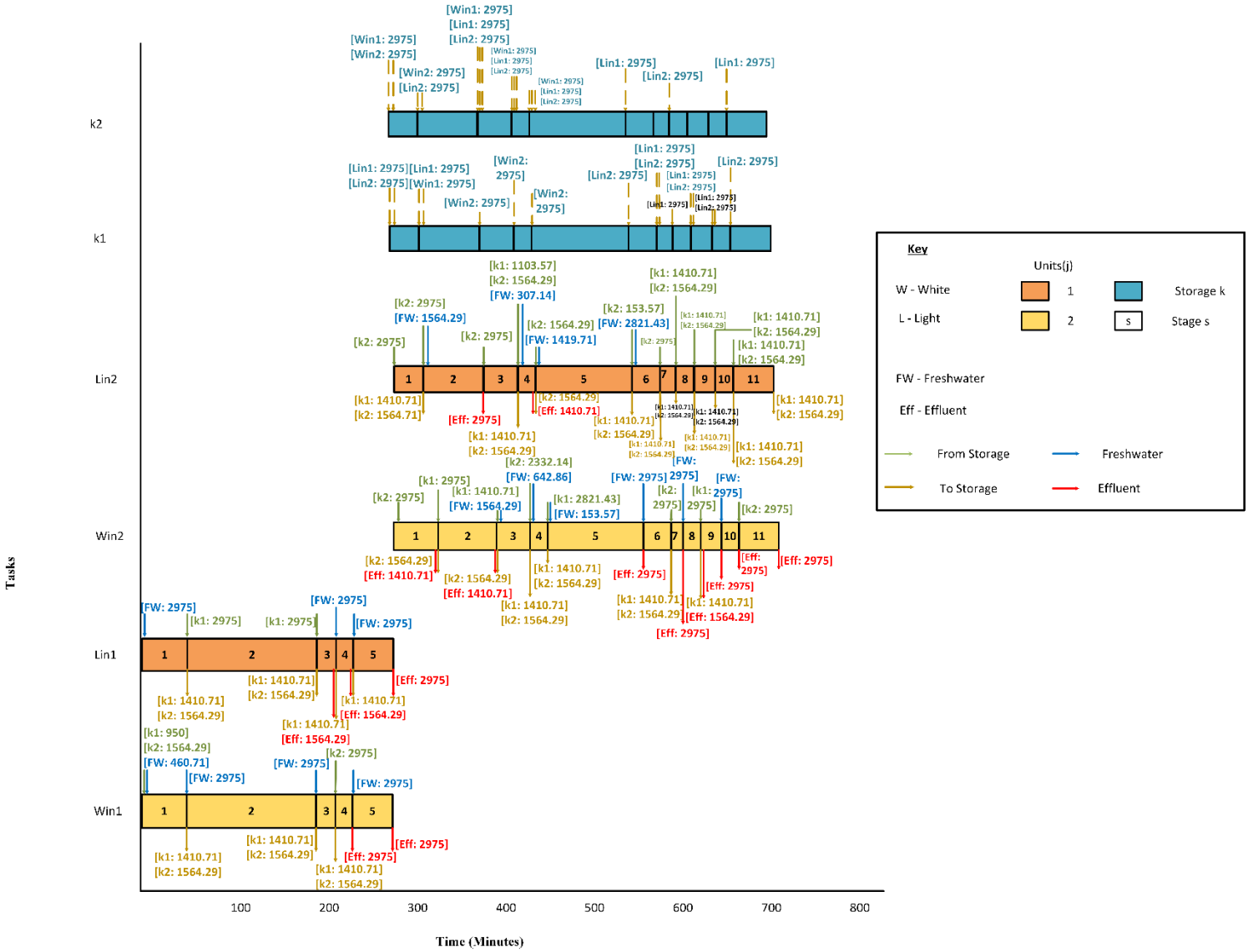


Figure 4.4. Gantt chart for Scenario A with storage tanks

Figures 4.5 and 4.6 illustrate the water schedules for Scenario B. Similar to the configuration in Figure 4.3, the water recycling in Figure 4.5 relies heavily on the immediate availability of

used water from preceding stages. However, a key feature of this scenario is the strategic alignment of the first stage of each batch with either the final stage of the preceding batch within the same unit or with the preceding batch from a different unit. This alignment is effective because the final stage of a batch typically produces water with the lowest concentration of contaminants, while the first stage demands the highest level of chemical deposition. This pairing allows for a logical and efficient matching of water streams based on quality and process requirements. Despite these efficiencies, the wastewater from the first stage of all tasks is discharged as effluent, as it contains the highest concentration of contaminants. This highly polluted water exceeds the allowable threshold for any subsequent stage's inlet concentration, rendering it unsuitable for reuse. This limitation highlights the constraints of operating without storage tanks, as the inability to buffer and treat water streams restricts the extent of recycling and leads to increased effluent generation.

Figure 4.6 illustrates the water usage patterns for Scenario B, where the white, light, and medium shades in the four units draw water from a combination of storage tanks, directly aligned stages, and freshwater supplementation at certain points. Notably, the light shades in units 3 and 4 commence processing later, allowing for better alignment with preceding stages and optimizing water reuse opportunities. The presence of storage tanks plays a crucial role as a buffer, enabling water to be stored when there are no immediate outlet-inlet stage matches for reuse. This flexibility ensures that water can be efficiently reused in later stages, even when direct alignment is not feasible. Additionally, it is noteworthy that the reused water does not require dilution in this scenario, further minimizing the need for freshwater.

The optimal network configuration for Scenario B incorporates 9 shared storage tanks, each with a finite capacity of 5 000 kg. These tanks facilitate water storage and distribution based on process demands. Similar to Scenario A, some stages in Scenario B rely on storage tanks for water supply, while others require freshwater for supplementation or dilution to meet quality requirements. Wastewater streams that cannot be reused due to quality constraints are either stored for later use or discharged as effluent, highlighting the integrated role of storage tanks in balancing water reuse, storage, and disposal across the water network.

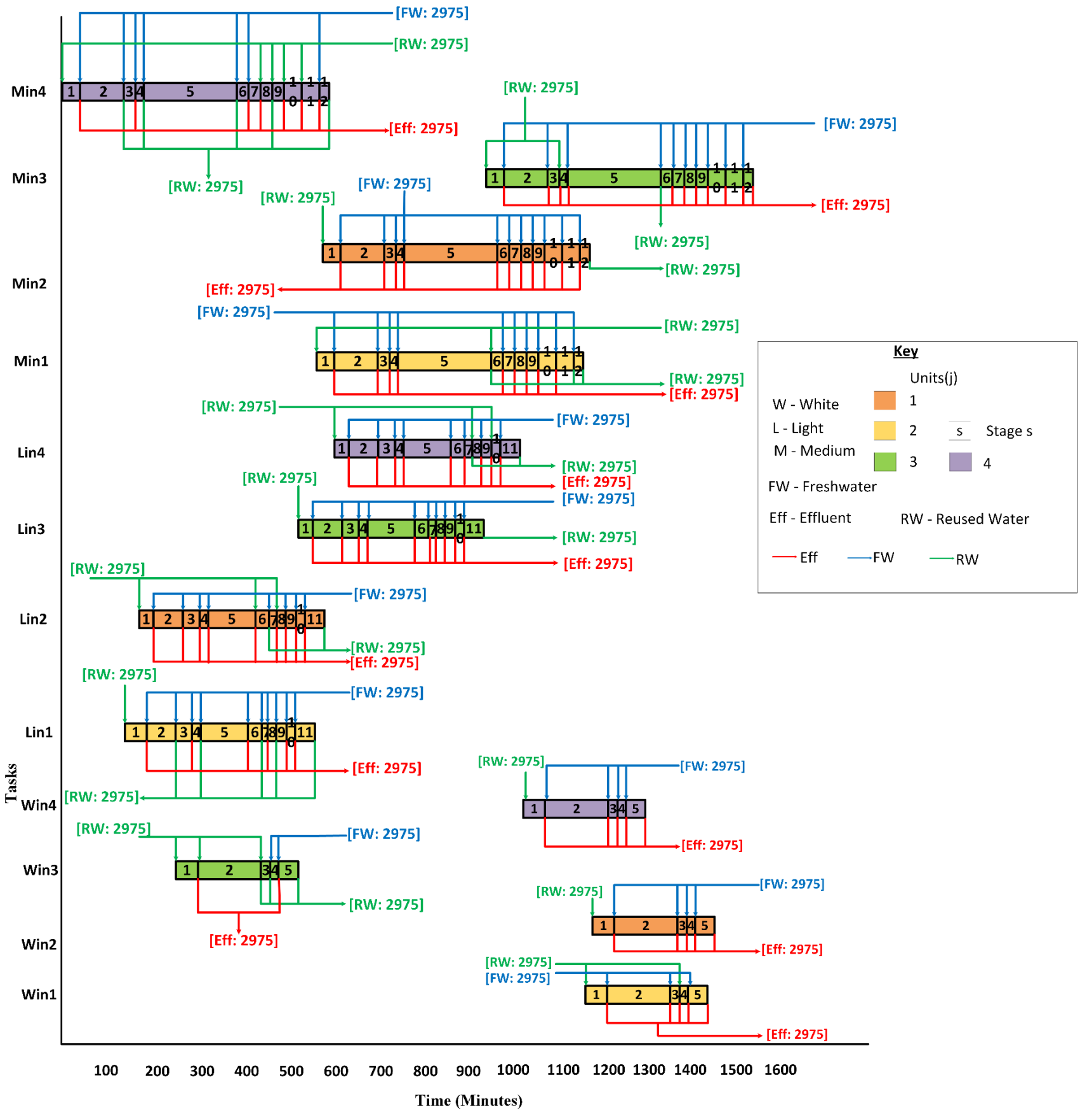


Figure 4.5. Gantt chart for Scenario B without storage tanks

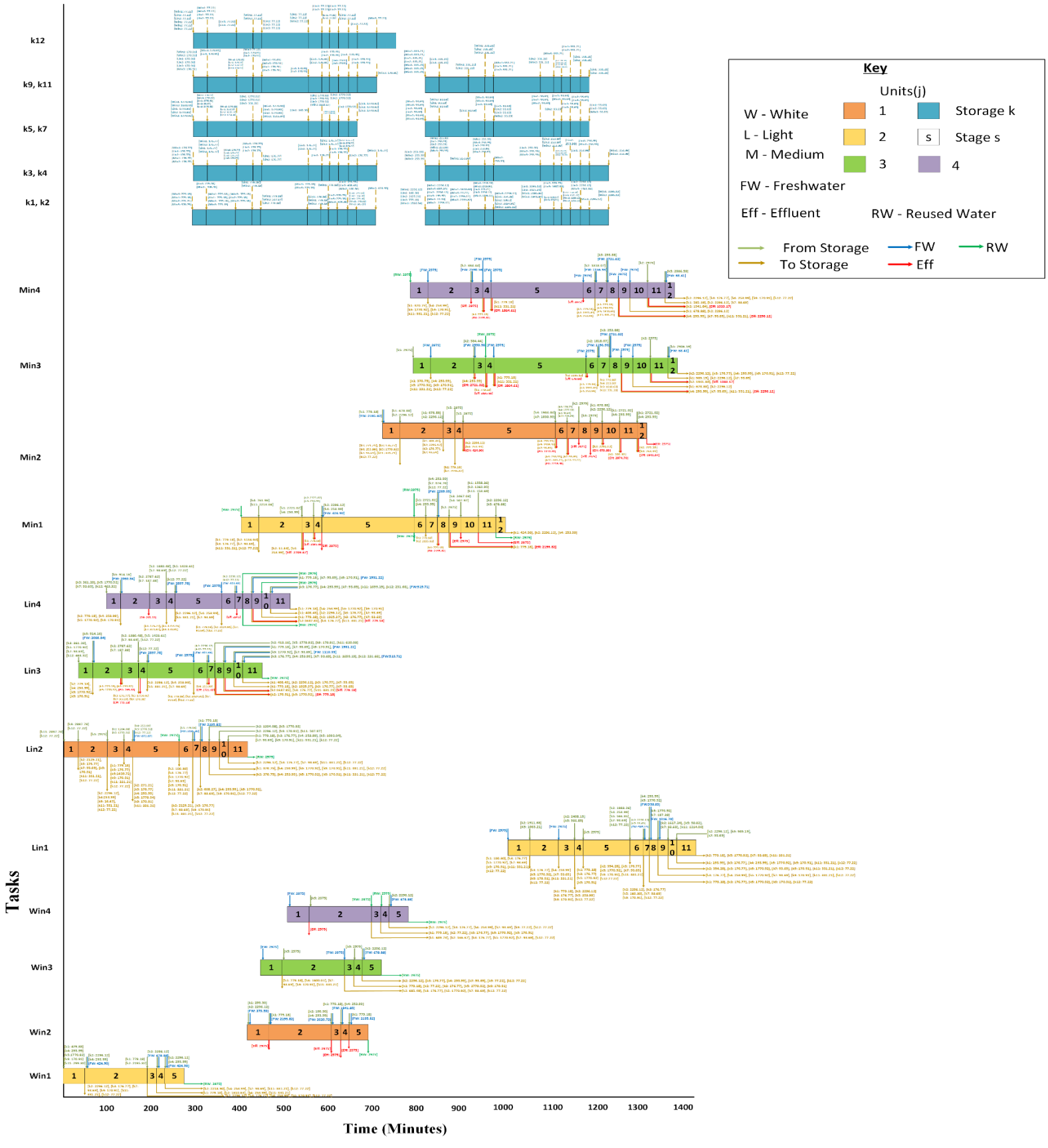


Figure 4.6. Gantt chart for Scenario B with storage tank

Table 4.3 provides a comparative analysis of the results obtained from the base case model and those obtained using the proposed formulation under two scenarios: without storage and with storage. The base case model includes constraints related to both scheduling and water balance but assumes that freshwater is the only water source for all operational needs. It excludes any opportunities for water reuse or recycling between processing units, meaning that all effluent from each stage is discharged as wastewater. The makespan, defined as the total duration of the process, is calculated as the sum of the time horizon and the slack variable. This parameter is a key metric for evaluating the efficiency of scheduling and water usage across the different scenarios.

Table 4.3. Comparative results for Case Study

Performance Metric	**Scenario A**			**Scenario B**		
	No water reuse	*Direct water reuse*	*Indirect water reuse*	*No water reuse*	*Direct water reuse*	*Indirect water reuse*
$\boldsymbol{\vartheta_{Slack}}$(in min)	-	106	0	-	197	77
MS (in min)	754	860	754	1 323	1 520	1 400
$\boldsymbol{mw_f}$(in kg)	95 200	77 350	35 700	333 200	255 850	107 628.8

The production campaigns in Scenarios A and B highlight the varying water requirements and process efficiencies under different water management strategies. In Scenario A, the campaign involves two batches each of products W and L i.e., 4 product batches in total. Scenario B, on the other hand, entails a more intensive campaign with four batches each of three products (W, L, and M) resulting in 12 product batches distributed across four units. For Scenario A, without any water reuse or storage, the process consumes 95 200 kg of freshwater to complete the campaign. Introducing water reuse without storage tanks reduces this requirement to 77 350 kg, representing an 18.6% reduction compared to the no reuse case. However, this improvement comes at the cost of a 106 minute increase in the makespan to accommodate stream matching between compatible stages. Incorporating storage tanks significantly enhances efficiency, reducing freshwater consumption to 35 700 kg, a 62.5% reduction, while maintaining the same makespan as the base case. In Scenario B, the absence of storage tanks

showed a 42.9% reduction in freshwater consumption, requiring 197 additional minutes in the makespan to enable stream matching. The inclusion of storage tanks further improves reuse opportunities, achieving a 60.4% reduction in freshwater demand. Notably, the makespan in this case increased to 1 400 minutes, reflecting the time required to optimize water distribution and reuse across the expanded water network.

Scenario B, with a higher number of product batches and greater process complexity, highlights the integration of a complete water network encompassing both direct and indirect water reuse scenarios. As the number of batches and products increases, so do the possible combinations and dependencies between stages, making water reuse opportunities more intricate. Storage tanks become even more valuable in this context, allowing for efficient handling of increased water demands while maintaining process flexibility. Despite the added scheduling complexity, the inclusion of storage tanks enables substantial reductions in freshwater consumption and effluent discharge by facilitating better coordination and reuse of water streams. However, it is important to acknowledge that incorporating storage tanks introduces additional costs, including capital investment, maintenance, and energy consumption for pumping and storing water. These costs must be weighed against the long-term benefits of reduced freshwater intake and wastewater disposal costs.

The complexity observed in Scenario B also highlights the limitations inherent in the current model when used to full-scale industrial environments. Although the model performs well under structured assumptions, its ability to represent the dynamic, nonlinear nature of real-world industrial operations such as fluctuating water quality, variable demand, and equipment constraints, is restricted. The intricate interdependencies between processing stages and water reuse pathways introduce additional scheduling challenges that significantly increase computational complexity. Scenario A provides a better grasp of these problems, as the model was evaluated on a smaller scale. These studies demonstrated that, while the model effectively minimized freshwater consumption and increased water reuse under simple settings, its performance was significantly dependent on the number of products, stages, and available storage capacity. Specifically, the evaluations highlighted that increasing product variety or limiting storage flexibility quickly resulted in infeasible or suboptimal solutions, indicating the model’s limited scalability.

To evaluate the effectiveness of the proposed model, a comparative analysis was carried out against established works by Nema et al. (2021), Croft et al. (2021) and Bhavsar et al. (2023), as summarized in Table 4.4. While each of these studies addresses critical aspects of scheduling and/or water minimization in multipurpose, multiproduct batch processing environments, they vary considerably in terms of their problem formulations, modelling frameworks, and optimization objectives. These differences offer a valuable context for comparing the proposed model's performance and suitability across a range of operational scenarios.

Table 4.4. Comparative analysis of recent studies on scheduling and water reuse in multipurpose multiproduct batch processes

Aspect	Nema et al. (2021)	Croft et al. (2021)	Bhavsar et al. (2023)	**Proposed Model**
Focus	Scheduling with water reuse using Finite Intermediate Storage (FIS)	Integrated scheduling and freshwater minimization via a sequential approach	Freshwater minimization through intentional waiting	Integrated scheduling and freshwater minimization through simultaneous optimization
Case Study Scale	Up to 14 stages, 6 batches, 4 product shades	Up to 14 stages, 6 batches, 4 product shades	3 stages, 4 product shades	Up to 11 stages, 3 batches, 3 product shades
Technique Used	GA- based simultaneous optimization; heuristic-based scheduling with FIS	Hybrid approach incorporating unit-specific continuous-time and global continuous-time approaches	GA- based approach	Unit-specific continuous-time
Freshwater Reduction	~ 76% reduction	~ 94% reduction	~ 59% reduction	~ 42.9% (without tanks), ~ 60.4% (with tanks)
Makespan Impact	-	-	~ 19% increase	~13% increase (without tanks),

				~ 6% increase (with tanks)
Computational Complexity	Low to Moderate (due to GA iterations)	Moderate to high (MILP scales with processing units)	Low to moderate (due to GA iterations)	High (evaluated at multiple scales)
Reuse Strategy	FIS with predefined paths	Centralized-reuse network; reuse optimization performed after scheduling	Interstage reuse through stage synchronization with tanks to align water-producing and consuming tasks	Direct and indirect reuse with optimization of reuse paths

Bhavsar et al. (2023) facilitated interstage water reuse by synchronizing production stages, thereby achieving significant reductions in freshwater consumption. However, their model did not explicitly explore scalability or indirect reuse networks. Nema et al. (2021) presented a water network optimization model, but the scheduling aspect was largely fixed, and their reuse was limited to indirect reuse through predefined loops. Croft et al. (2021) delivered the highest freshwater reduction (94%) through a hybrid scheduling approach that enhanced order-level efficiency and batching. Additionally, they achieved notable makespan reductions, which sets their model apart in terms of scheduling performance. However, their model did not explicitly incorporate direct water reuse, which may limit its adaptability in facilities with constrained storages. As a result, while effective in freshwater savings, the approach may not translate directly to settings where storage infrastructure is limited.

As illustrated in Table 4.4, each of the reviewed models offer valuable contributions to water reuse optimization. Nevertheless, the proposed framework, despite its computational demands, offers a more holistic and flexible solution by simultaneously optimizing reuse strategies and production makespan. It accommodates both direct and indirect reuse, adapts to varying plant configurations, and demonstrates robust flexibility. This integrated approach makes the model suitable for practical implementation in industries that seek to improve operational efficiency while reducing freshwater consumption.

4.4 References

Bhavsar, A., Srinivasan, B., & Srinivasan, R. (2023). Freshwater minimization in multistage multiproduct batch processes through intentional waiting. *Chemical Engineering Research and Design*, *196*, 204–213. https://doi.org/10.1016/j.cherd.2023.06.039

Buabeng-Baidoo, E., Mafukidze, N., Pal, J., Tiwari, S., Srinivasan, B., Majozi, T., & Srinivasan, R. (2017). Study of water reuse opportunities in a large-scale milk processing plant through process integration. *Chemical Engineering Research and Design*, *121*, 81–91. https://doi.org/10.1016/j.cherd.2017.02.031

Croft, J., Alam, M. N., Machakaire, N., Srinivasan, B., Srinivasan, R., & Majozi, T. (2021). Synthesis of an Optimal Schedule and Water Network for a Multipurpose Multiproduct Textile Industry through a Sequential MILP-NLP Technique. *Industrial and Engineering Chemistry Research*, *60*(45), 16363–16377. https://doi.org/10.1021/acs.iecr.1c02661

Floudas, C. A., & Pardalos, P. M. (2009). *Encyclopedia of Optimization*. Springer.

Grossmann, I. E. (2012). Advances in mathematical programming models for enterprise-wide optimization. *Computers and Chemical Engineering*, *47*, 2–18. https://doi.org/10.1016/j.compchemeng.2012.06.038

Grunow, M., & Lehmann, M. (2002). Campaign planning for multi-stage batch processes in the chemical industry. In *OR Spectrum* (Vol. 24). Springer-Verlag.

Nema, A., Srinivasan, B., Majozi, T., & Srinivasan, R. (2021). A simple strategy to maximize water-reuse in multistage, multiproduct batch processes. *Chemical Engineering Research and Design*, *168*, 327–339. https://doi.org/10.1016/j.cherd.2021.02.012

Zhou, Y., Wang, J., Zhang, P., Wang, P., Lu, Y., & Zhang, J. (2020). Research on dyeing workshop scheduling methods for knitted fabric production based on a multi-objective hybrid genetic algorithm. *Measurement and Control (United Kingdom)*, *53*(7–8), 1529–1539. https://doi.org/10.1177/0020294020944947

Chapter 5

Limitations

5.1 Introduction

The water network scheme developed in this work was verified and validated against a real-world industrial process, ensuring its relevance and applicability. During the implementation phase, several challenges arose, primarily due to the inherent complexity and structural characteristics of the model. These factors contributed to computational hurdles, including prolonged solution times and difficulties in achieving convergence. This chapter highlights the key limitations encountered in the proposed formulation, examining their impact on model performance and industrial applicability. Furthermore, the discussion extends to potential strategies for future improvements.

5.2 Limitations

5.2.1 Computational Intensity

The integration of the water network model into the production scheduling model significantly increased the scale and complexity of the optimization problem, leading to computational limitations that hindered the ability of the solver to converge efficiently on the largest scale of the industrial case study. The primary challenge stemmed from the addition of nonlinear constraints and a large number of binary variables, which exponentially increased the computational burden. Nonlinear constraints, particularly those arising from water reuse, contaminant balance equations, and storage, introduced non-convexities that made the problem inherently more difficult to solve (Grossmann, 2012).

Although CPLEX is able to effectively handle the MILP sub-problems, the NLP sub-problems solved by CONOPT3 become increasingly complex as the number of nonlinearities and

decision variables grows. Large-scale problems often encounter issues such as excessive computation times, memory limitations, and failure to find feasible solutions within acceptable time frames. This is particularly evident when non-convexities lead to multiple local minima, increasing the likelihood of the solver getting trapped and failing to identify the global optimum (Floudas, 2000). Moreover, the addition of the water network model introduced a large number of interdependencies between production scheduling and water reuse/recycling operations. These interdependencies increased the dimensionality of the problem, as variables such as water intake and outtake, contaminant concentrations, and tank storage capacities had to be simultaneously optimized alongside production tasks. The expanded solution space made it difficult for the solver to navigate efficiently, particularly in cases with high batch numbers and multiple units, as observed in the industrial case study.

Computational challenges in large-scale optimization problems can be mitigated through techniques like convexification, which includes methods such as Glover's transformation and McCormick envelopes (discussed in section 2.6). These methods are used to reformulate non-convex problems into convex ones, making them easier to solve (Tawarmalani & Sahinidis, 2002). Additionally, advanced solution strategies such as problem decomposition and hybrid approaches can be employed. In these strategies, the integrated model is broken down into smaller sub-problems, such as production scheduling and water network optimization. These are then solved iteratively or sequentially to reduce the computational burden.

In addition to addressing computational challenges, it is important to critically examine the underlying assumptions embedded within the model. For instance, assumptions such as perfect synchronization of parallel processing lines are often introduced to simplify model formulations and improve solvability. However, these simplifications may limit the model’s applicability to real-world scenarios where variability and uncertainty are inherent. In practice, such assumptions can be relaxed through more flexible formulations that allow variability of process timing, uncertain demand, or dynamic operating conditions. Approaches such as stochastic techniques or robust optimization can be employed to account for these complexities, thereby improving the model’s practical relevance and robustness.

5.2.2 Further Water Saving Potentials

The absence of a water regeneration process in the developed water network model significantly limited its ability to explore further water-saving opportunities. Regeneration units are designed to treat and recycle water streams, making them suitable for reuse by reducing contaminant concentrations to acceptable levels. Excluding them made the model rely solely on direct and indirect water reuse, reducing the flexibility of the water network to accommodate higher contaminant loads, especially in scenarios where water reuse alone could not meet the quality requirements of subsequent processes.

In real-world scenarios, regenerators enable higher levels of water recycling by treating effluent to meet the specific contaminant concentration thresholds of individual processes. This omission, therefore, meant that the optimization could not fully evaluate the trade-offs between the operational costs of regeneration and the potential freshwater savings (Bagajewicz, 2000). The computational challenges associated with solving large-scale problems also resulted in the inability to include a regenerator in the water network model. Integrating a regenerator introduces additional variables and nonlinear constraints, such as treatment capacity, efficiency, and cost equations, which significantly increase the complexity of the optimization problem. These additions transform the problem into a more complex MINLP formulation, requiring greater computational resources and time for convergence (Floudas, 2000).

5.3 References

Bagajewicz, M. (2000). A review of recent design procedures for water networks in refineries and process plants. In *Computers and Chemical Engineering* (Vol. 24). www.elsevier.com/locate/compchemeng

Floudas, C. A. (2000). *Deterministic Global Optimization: Theory, Methods and Applications*. Springer Science & Business Media.

Grossmann, I. E. (2012). Advances in mathematical programming models for enterprise-wide optimization. *Computers and Chemical Engineering*, *47*, 2–18. https://doi.org/10.1016/j.compchemeng.2012.06.038

Tawarmalani, M., & Sahinidis, N. V. (2002). *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming* (P. Pardalos, Ed.). Springer-Science+Business Media, B.Y.

Chapter 6

Conclusion and Future Work

This work addresses the optimal scheduling and water minimization for a multipurpose multiproduct textile plant using a simultaneous approach. A unit-specific event-based MILP model was developed, resulting in optimal production sequences across multiple processing units while accounting for each product's scheduling requirements. It minimized the overall makespan thereby satisfying the demand in a time-efficient manner. A water reuse model was then created and integrated into the overall framework, allowing for both direct and indirect reuse of water across compatible stages and units. The model captured opportunities for freshwater substitution through internal recycling, which was facilitated by the presence of storage tanks. In the final integrated model, a unit-specific event-based formulation was employed to achieve the optimal water network, incorporating a time penalty function to account for the scheduling trade-offs inherent in water reuse decisions.

The resultant MINLP formulation was applied to an industrial case study from a multistage multiproduct textile plant and solved using the CONOPT3 solver. The developed water network demonstrated feasible water savings, indicating potential reductions in environmental impact for multipurpose multiproduct batch facilities, along with significant economic benefits from the optimized makespan. However, the computational complexity of synthesizing such a comprehensive water network alongside production scheduling underscores its primary utility during the design phase of batch plants. This dual optimization of time and water usage emphasizes the importance of balancing environmental sustainability with industrial practicality.

The current scope does not include storage optimization, economic analysis, or wastewater regeneration, leaving room for practical extensions. Future work could, therefore, explore the optimization of storage tank capacities considering factors such as tank cost and tank size to

avoid unnecessary costs from overdesign or underdesign. The proposed model prioritizes minimizing water usage and makespan but does not incorporate an economic analysis of the associated trade-offs. Future studies could hence assess the economic impact of incorporating storage tanks, and the cost savings associated with reducing freshwater consumption and effluent discharge. Finally, exploring alternative methods and solvers that enable the model to solve optimally within a reasonable timeframe could improve its scalability, allowing for further exploration of water-saving opportunities, such as integrating wastewater regeneration into the water network.