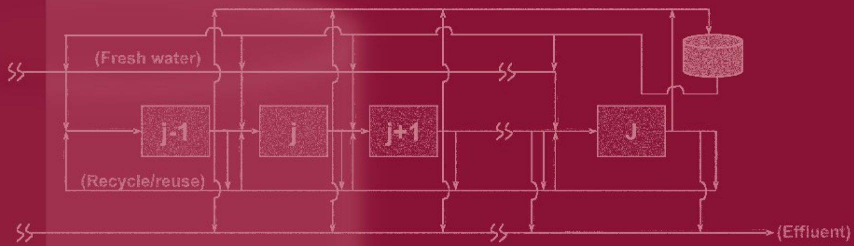
Thokozani Majozi



Batch Chemical Process Integration

Analysis, Synthesis and Optimization

Springer

Batch Chemical Process Integration

Thokozani Majozi

Batch Chemical Process Integration

Analysis, Synthesis and Optimization

Springer

Prof. Dr. Thokozani Majozi
Department of Chemical Engineering
University of Pretoria
Lynnwood Road
0002 South Africa
thoko.majozi@up.ac.za

ISBN 978-90-481-2587-6 e-ISBN 978-90-481-2588-3
DOI 10.1007/978-90-481-2588-3
Springer Dordrecht Heidelberg London New York

Library of Congress Control Number: 2009939334

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

‘God, Almighty, advance me in knowledge, understanding and humility’ To all those who have given and all those who continue to give my life a sense of purpose.

Foreword

Over the past three decades, process integration has evolved as a holistic approach to design and operation which emphasizes the unity of the tackled systems. The overwhelming majority of research publications and textbooks in the field have focused on continuous systems. This has been the case for two main reasons. First, until recently, most of the sizable process industries have been designed to operate in a near steady-state and continuous mode. This is changing given the need to produce a number of specialty chemical to address variable market needs, the increasing level of flexibility, and the emergence of new industrial sectors (e.g., biorefineries) that favor batch operations. Second, process integration techniques for unsteady-state operations are more challenging than those for steady-state operations. As such, contributions to the field of batch process integration have come from a limited number of researchers. Such contributions have invoked advanced concepts in process design, operation, and scheduling, network synthesis and analysis, and some graphical but largely mathematical programming techniques. Hence, these contributions have been read and utilized by a select few experts. There has been a clear literature gap. Therefore, it was with great delight that I learned about Prof. Thokozani Majozi's project to overcome this literature gap by introducing this textbook that addresses batch chemical process integration. Having followed Prof. Majozi's exciting work in the field, I was convinced that the product will be superb. Indeed, now that the book is complete and that I had the privilege of reviewing it in full, I am thrilled that such an outstanding contribution is now available to researchers, students, and practicing engineers. The book is very well written and gradually introduces key concepts in batch process integration including the necessary background in mathematical programming, network representation, and operational concepts. The book also emphasizes the conceptual framework behind many of the mathematical formulations and focuses on the insights that drive the design, operation, and scheduling strategies. The book is loaded with examples that streamline the concepts and facilitate the learning process. There is a nice spectrum of applications ranging from basic manufacturing to waste reduction (primarily water management and wastewater minimization) to heat integration. This is a much-needed and highly-valued book that will open the door for many readers to learn the fundamentals and application of batch process integration.

Dallas, TX Mahmoud El-Halwagi

Preface

Research in batch processes only received heightened interest in the last 2 to 3 decades. Most of the work in published chemical engineering literature tends to focus on continuous processes at steady state. This occurrence dovetails with the evolution of the chemical industry as well as the dynamics of the global markets since the dawn of industrial revolution. From the late nineteenth to the mid-twentieth centuries, global markets were characterised by reasonable stability and crafted on bulk demand and mass production which favoured continuous processes. Demand for small volume high value added products constituted a very small fraction in that era. This pattern, however, began to change drastically in the latter part of the last century, with major markets displaying high levels of volatility that required processes amenable to sudden changes. Batch processes are ideally suited for this situation. Consequently, research in batch process scheduling began in earnest from the mid to the late 1970s. Scheduling, which is aimed at capturing the essence of time, is the cornerstone of all batch related activities, including *Process Integration*.

Process integration was developed and rose to prominence during the energy crisis of the 1970s in the form of Pinch Technology. The latter proved to be the breakthrough in energy optimization and sustainable design. It advocates the exploration of maximum energy recovery within the process through process–process heat exchange prior to resorting to external utility requirements. Its strength lies in the ability to set energy targets before commitment to design. Moreover, its graphical nature allows the designer to guide the optimisation process, which is not necessarily the case with mathematical approaches. This finally results in an energy efficient heat exchanger network (HEN). It still remains one of the major advances in chemical engineering even today. However, this contribution was aimed at continuous processes at steady state and ignored the impact of time dependent interventions as traditionally encountered in batch processes. This omission was not seen as a major drawback in process integration within the chemical engineering community, since continuous processes have largely been perceived to be much more energy intensive than their batch counterparts. The concept of process integration was later extended to mass exchanger networks with the ultimate goal of waste minimisation in 1989 where it also proved to be a major contribution. Again, the focus at the early stages of this advancement was on continuous rather than batch processes for similar reasons.

Whilst methods for scheduling batch processes advanced steadily throughout the 2 quarters of the last century, process integration aspects pertaining to energy and waste minimisation still remained largely isolated from the mainstream research. There were indeed a few contributions in this regard, but their impact remained minimal for one main reason. They were largely a direct adaptation of the techniques developed for continuous processes, which meant that the time dimension had to be directly or indirectly suppressed in the analysis, thereby resulting in somehow inaccurate results. Stringent environmental legislation and the growth of batch processes in the industrial sector have necessitated research on the development of process integration techniques that are particular to batch processes. Since the beginning of this century, significant advances have been made in this regard. It is becoming clear, however, that batch processes, unlike continuous processes, are more amenable to mathematical than graphical analysis. This situation arises mainly from the added time dimension that makes it difficult to confine batch process analysis to 2 dimensions as traditionally encountered in graphical methods.

This textbook presents a comprehensive overview of some of the milestones that have been achieved in batch process integration. It is largely based on mathematical techniques with limited content on graphical methods. This choice was deliberately influenced by the observation made in the foregoing paragraph, i.e. in order to handle time accurately mathematical techniques seem to be more equipped than their graphical counterparts. The book is organised as follows.

- Chapter 1 gives an overview of batch processes.
- Chapter 2 introduces the reader to the basis of all the mathematical techniques presented in this textbook. The mathematical techniques are founded on a recipe representation known as the state sequence network (SSN), which allows the use of states to dominate the analysis thereby reducing the binary dimension.
- Chapter 3 presents a synthesis technique for multipurpose batch plants and further introduces an unexplored operational philosophy so called Process Intermediate Storage (PIS) operational philosophy.
- Chapter 4 presents a technique for wastewater minimisation in batch plants with single contaminants.
- Chapter 5 addresses the optimum design of intermediate water storage in multiproduct and multipurpose batch plants.
- Chapter 6 presents a technique for wastewater minimisation in multipurpose batch plants characterised by multiple contaminants.
- In Chapter 7 a mathematical technique that takes into account presence of multiple reusable water storage vessels is presented.
- A near-zero effluent approach is presented in Chapter 8. In particular, this chapter focuses on a special class of batch plants wherein water is a major constituent of the final product.
- Chapter 9 presents a mathematical technique for wastewater minimisation through exploitation of idle processing units, which is a unique and largely inherent feature of batch plants.

- Heat integration is addressed in Chapter 10 and 11. Chapter 10 focuses on direct heat integration whilst Chapter 11 on indirect heat integration.
- Lastly, Chapter 12 presents graphical techniques in wastewater minimisation of batch processes as well as a brief comparison between graphical and mathematical techniques. The comparison aims to highlight the necessity of time in batch plants.

University of Pretoria,
South Africa

Thokozani Majozi

Acknowledgments

This work could not be brought to life without the help of several people who gave guidance, assisted with research material and kept me focused throughout the writing of this textbook. Most of these people are my current and former students who have worked tirelessly in understanding batch processes. First and foremost, I would like to thank my recent PhD graduate student Dr Jacques Gouws with whom we shared a very common vision from the onset of his research. All the seeds of ideas I shared with him landed on a fertile ground and grew rapidly to applicable methodologies that have been adopted and implemented for industrial benefit. Perhaps in the same league is Mr Thomas Pattinson who put significant effort in the development of the formerly unexplored operational philosophy in batch plants, viz. Process Intermediate Storage (PIS) operational philosophy.

I am sincerely grateful to many of my recent students who had to edit some of the sections of this book whilst keeping with my demands on their research. In no particular order, at the fore of my mind in this regard are, Miss Bola Adekola, Miss Jane Stamp, Mr. Tim Price, Mr. Donald Nonyane, Mr. Vhutshilo Madzivhandila, Mr Vincent Gololo and Mr Esmael Reshid. Indeed, I am deeply indebted to my former supervisor, Dr Frank Zhu, who instilled in me love for batch operations and highlighted the urgency of generating appropriate techniques for these operations. He is also the one who introduced me to the world of mathematics and always mentioned that in every step of the way I must never compromise beauty for there is no permanent place in this world for ugly mathematics – confirming the famous mathematician Godfrey Harold Hardy's observation.

I am also deeply indebted to the following people who have offered guidance in various roles; as mentors, as friends and as family. Professor Ferenc Friedler, the Dean of Information Technology at the University of Pannonia in Hungary who has been my longstanding mentor, since I met him in 2000 in Los Angeles. He has always been the source of intellectual support and a reliable sounding board to most of my ideas. My former supervisors, Professor Chris Buckley and Mr Chris Brouckaert for their faith in me during my very early stages of research. Professor Toshko Zhelev from the University of Limerick in Ireland who has always had time to listen, criticise and encourage constructively since our first encounter in 1997 in South Africa. His patience with me still remains unprecedented. Professor Mahmoud El-Halwagi, the author of the foreword to this book, who I did not have to meet in person to derive inspiration from him. My true interest in Process Integration started whilst reading his book on 'Pollution Prevention through Process Integration' in 1997. I read each of its 314 pages with great passion. Mr Thanda Sibisi, my former science and mathematics teacher, who instilled in me the love and understanding for these two wonderful subjects and further advised me to pursue chemical engineering at my most desperate moment. My wife Bongiwe 'Mabongi' Dube and my children, Ntsika and Olwanda (the bunnies), who have been very understanding of my absence even when I am physically in their midst. Lastly, but certainly not the least, my dear parents who were beginning to understand the challenges of batch process integration towards the end of writing this book.

Let me not forget the mention of those who have provided fuel for my research vehicle to propel to the fore – the financial sponsors. Predominant among these are Dr Sibisi, the President and CEO of the Council for Scientific and Industrial Research (CSIR), Mr. Cyril Gamede, the Operations Director of African Explosives Limited (AEL) and Mr. Leon Kruger, the Operations Director of Johnson and Johnson (Pty) Ltd. My heartfelt thanks also go to the Water Research Commission (WRC) of South Africa.

Contents

Chapter 1
Introduction to Batch Chemical Processes

Overview Batch processes are mostly suited to low volume high value added products that are usually characterised by common recipes, which render them amenable to sharing of equipment units. Due to their intrinsic adaptation to sudden changes in recipe, they are processes of choice in volatile or unstable conditions that have become regular in global markets. This chapter provides the background information on batch chemical processes, which constitutes the basis for the forthcoming chapters. Only the essential elements of batch plants are captured with references, where necessary, to further sources of information for the benefit of the reader.

1.1 Definition of a Batch Process

Any process which is a consequence of discrete tasks that have to follow a predefined sequence from raw materials to final products is a batch process. This predefined sequence is commonly known as a recipe. The primary features of any comprehensive recipe are the quantities of materials that have to be processed by individual tasks as well as the duration of each task within the recipe. The secondary features are the operating conditions of the various tasks, and in less common circumstances, the locality or geographic position of the task at hand. In processes wherein safety is of great concern, it might be necessary to perform a particular task in a designated area equipped with relevant safety features.

In reality, it is the *discreteness* of tasks that differentiates batch processes from their continuous counterparts. To illustrate, Fig. 1.1a shows a typical batch reactor with all the tasks comprising the entire batch reaction. On the other hand, Fig. 1.1b depicts a typical continuous reactor at steady-state. The discreteness of tasks in Fig. 1.1a is evident, which is not the case in Fig. 1.1b. Consequently, it is fair to deem batch processes 'distributed in time', whilst continuous processes, at steady-state, are 'frozen in time'.

Another illustration for the distinction between batch and continuous process is depicted in Fig. 1.2. The discreteness of tasks that characterise a batch process is evident in Fig. 1.2a. The use of storage becomes necessary when the completion of

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_1,

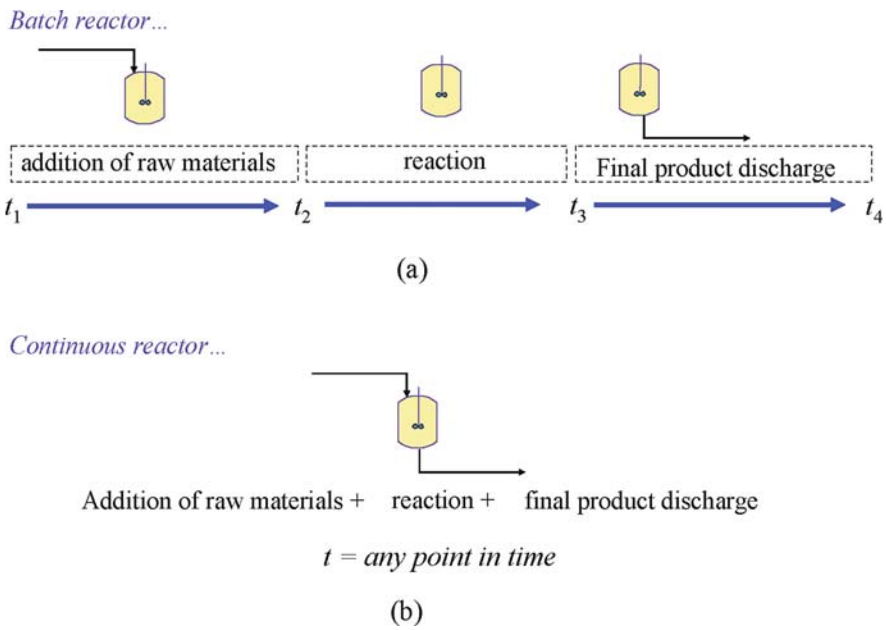


Fig. 1.1 (**a**) Batch vs (**b**) continuous reaction

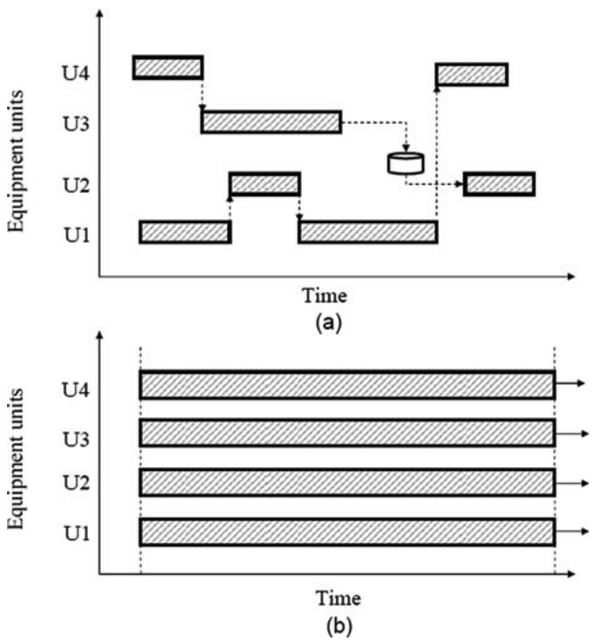


Fig. 1.2 (**a**) Batch vs (**b**) continuous process operation

one task does not coincide with the commencement of the next task in the recipe. This condition, which is a regular occurrence in batch plants, mainly arises from the unavailability of processing equipment units. A processing equipment unit is only available once it has completed all the previous tasks. Moreover, the inclusion of storage is invariably concomitant with flexibility in the operation of the plant, which is likely to increase throughput or even minimize the makespan. On the other hand, over a given time horizon, a continuous process, at steady state, would exhibit the behaviour shown if Fig. 1.2b. The feature of batch processes shown in Fig. 1.2a, commonly known as the Gantt chart, implies that a proper design of the batch plant requires exact treatment of time, since each task is time dependent. In a nutshell, the optimum capacities of each of the equipment units are inextricably linked to the optimum schedule. Consequently, in order to properly design a batch plant, one cannot overlook nor disregard the underlying schedule. Again, this does not feature in continuous operations.

1.2 Operational Philosophies

In the analysis, synthesis and optimization of batch plants complexity arises from the various operational philosophies that are inherent in time dependent processes. In a situation where the intermediate is allowed to wait in the same unit from which it is produced until the next unit is available, the operational philosophy is commonly known as no intermediate storage (NIS) operational philosophy. This philosophy is depicted in Fig. 1.3. NIS operational philosophy is usually adopted if operational space is of essence, since intermediate storage tanks can occupy considerable area.

Intermediate storage is usual the most obvious means to debottleneck the batch process and introduce flexibility. In general, 3 operational philosophies take advantage of intermediate storage in rather distinct ways. The first is the finite intermediate storage (FIS) operational philosophy, depicted in Fig. 1.4. FIS operational philosophy stipulates that the intermediate is stored prior to the next step in the recipe. The storage of intermediate material could be due to capacity or time related issues. If the capacity of the source of the intermediate is larger than the capacity of the sink, storage might prove necessary so as to contain all the material within the process. On the other hand, as mentioned earlier in this chapter, if the completion time of

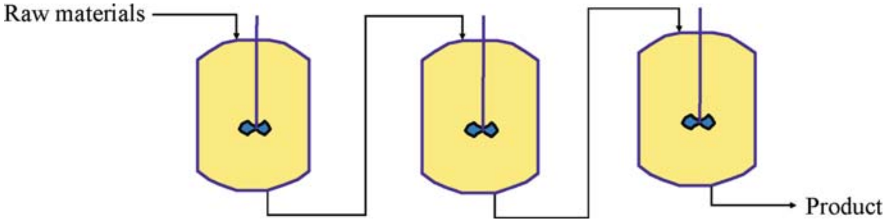


Fig. 1.3 No intermediate storage (NIS) operational philosophy

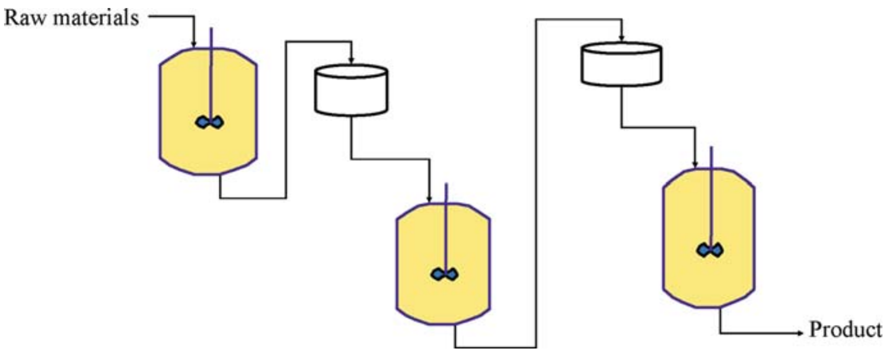


Fig. 1.4 Finite intermediate storage (FIS) operational philosophy

one task does not coincide with the starting time of the subsequent task, an intermediate storage unit can be used to debottleneck the process. The finiteness of storage, as the name of this philosophy suggests, emanates from the fact that the availability of storage capacity is not guaranteed. This implies that there could be instances within the time horizon of interest when the storage unit is filled to its capacity and consequently unavailable for further storage.

The second operational philosophy that exploits intermediate storage is the unlimited intermediate storage (UIS) operational philosophy. This philosophy is similar to FIS philosophy, except that the availability of storage is always guaranteed. The implication thereof is that whenever the intermediate material is produced it can immediately be stored without limitations or constraints on storage capacity. In practical terms, this can be achieved if the capacity of storage is too large compared to the capacity of production units as shown in Fig. 1.5.

The third philosophy that makes use of intermediate storage is common intermediate storage (CIS) operational philosophy, shown in Fig. 1.6. CIS philosophy involves the sharing of storage by various tasks within the batch plant. Needless

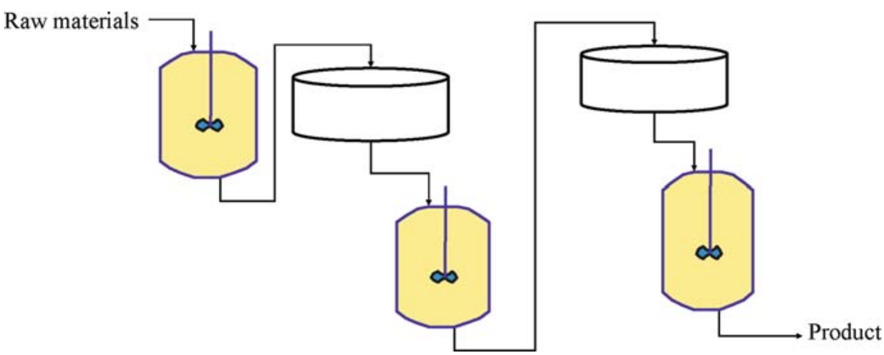


Fig. 1.5 Unlimited intermediate storage (UIS) operational philosophy

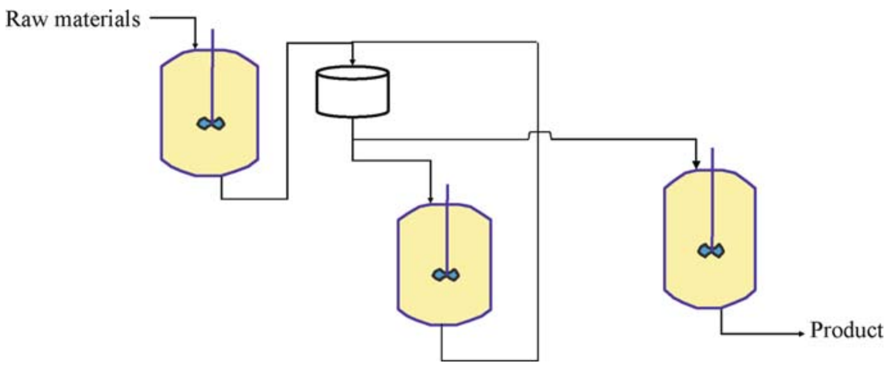


Fig. 1.6 Common intermediate storage (CIS) operational philosophy

to mention, this philosophy requires stringent measures to ensure that product integrity is not compromised. For example, the storage tank might have to be washed thoroughly between different types of materials, thereby resulting in effluent and associated cost.

The other operational philosophies that are generally encountered are the mixed intermediate storage (MIS), zero-wait (ZW), finite wait (FW) as well as the unlimited wait (UW) operational philosophies. The MIS philosophy is encountered in a situation where at least 2 of the aforementioned operating philosophies coexist in one process. It is indeed very seldom in most practical applications to have only one philosophy throughout the operation. A combination of different philosophies in often the case.

The ZW, FW, and UW are in most instances a consequence of product stability. In a situation where the intermediates are unstable, it is always advisable to proceed with the subsequent step(s) in the recipe as soon as the intermediates are formed, hence the ZW operational philosophy. Due to its nature, ZW does not require any dedicated storage for the intermediates and could be depicted by a flowsheet similar to that shown in Fig. 1.3. On the other hand, the intermediate could be partially stable and only commence decomposition after a certain period. In this case storage time has to be finite in order to prevent formation of unwanted material, hence the FW operational philosophy. The UW operational philosophy is applicable whenever the intermediates are stable over a significantly longer time than the time horizon of interest. In both FW and UW operational philosophies, storage of intermediates can either be within the processing equipment or dedicated storage unit.

1.3 Types of Batch Plants

Batch chemical processes are broadly categorised into multiproduct and multipurpose batch plants. In multiproduct batch plants, each produced batch follows the same sequence of unit operations from raw materials to final products. However, the

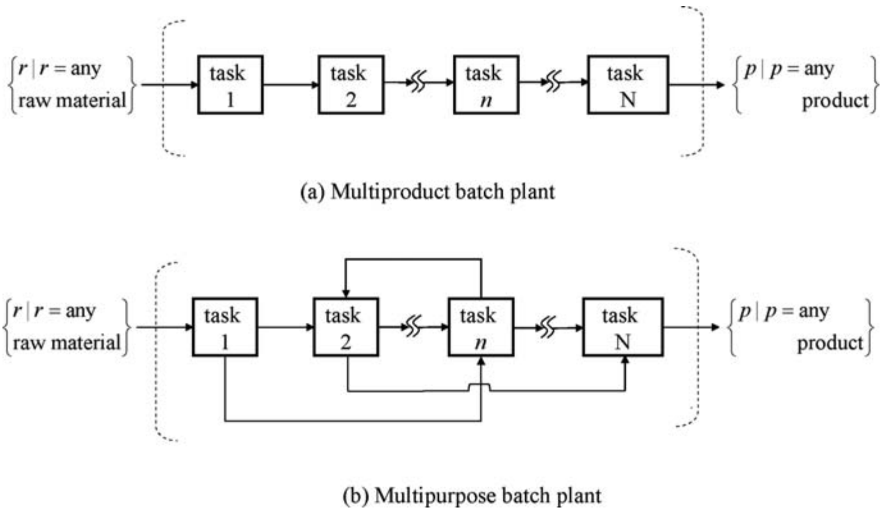


Fig. 1.7 (**a**) Multiproduct and (**b**) multipurpose batch plants

produced batch need not belong to the same product and the duration of tasks corresponding to different products can vary. Consequently, multiproduct batch facilities are ideally suited to products with identical and fixed recipes as shown in Fig. 1.7a. If the recipes of the products involved vary from one batch to another, multipurpose batch facilities, depicted in Fig. 1.7b, tend to be the ideal choice. The variation in recipes for the different batches does not necessarily mean the variation in products. In other words, the same product can have different recipes. As a result, multipurpose batch facilities are appropriate in the manufacture of products that are characterized by variations in recipes.

It is evident from the foregoing description and diagrams shown in Fig. 1.7a, b that multipurpose batch chemical plants are more complex than multiproduct batch plants. This complexity is not only confined to operation of the plant, but also extends to mathematical formulations that describe multipurpose batch plants. Invariably, a mathematical formulation that describes multipurpose batch plants is also applicable to multiproduct batch plants. However, the opposite is not true. It is solely for this reason that most of the effort in the development of mathematical models for batch chemical plants should be aimed at multipurpose rather than multiproduct batch plants.

1.4 Capturing the Essence of Time

Since they are comprised of time dependent tasks, it is paramount that time is addressed in an almost exact manner in describing batch chemical processes. Any attempt that seeks to bypass or override this fundamental feature of batch processes is likely to fail at worst and be too inaccurate at best. The implication thereof is that methodologies that are meant for continuous operations in which the time dimension is overridden cannot be directly applied in batch operations. Capturing the essence of time is arguably the most challenging aspect of batch chemical process integration.

In published literature there exist 3 types of methods in which the influence of time is handled. The first type involves the use of time average models (TAMs) which ultimately treat batch plants as pseudo-continuous operations. As aforementioned this cannot yield results that are a true representation of reality insofar as it attempts to describe batch processes. The second type treats time as a fixed parameter that is known a priori with no opportunity for change within the time horizon of interest. The main drawback of this approach is that true optima associated with treating time as a variable rather than a parameter are likely to be overlooked. The traditional graphical targeting techniques on which process integration is founded are highly amenable to these 2 types of methods, since they treat time as a suppressed dimension in the analysis. This consequently allows the analysis to be confined to 2 dimensions, which is an inherent feature of most graphical techniques.

The third type of methods treats time in an exact manner by allowing it to vary in search of a true optimum. Worthy of mention at this stage is that none of the graphical techniques bears this crucial capability in batch chemical plants. Ultimately, mainly mathematical techniques are used in order to treat time exactly. This assertion is further justified later in this textbook when graphical techniques are compared to mathematical techniques using a typical problem.

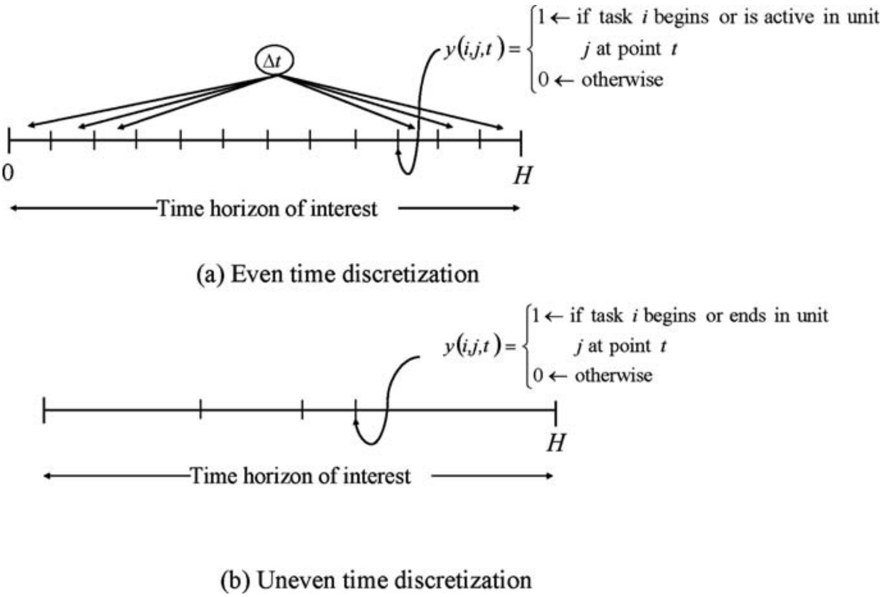


Fig. 1.8 (**a**) Even and (**b**) uneven time discretization

Needless to mention, the exact capturing of time presents further challenges in the analysis. Fundamentally, a decision has to be made on how the time horizon has to be represented. Early methods relied on even discretization of the time horizon (Kondili et al., 1993), although there are still methods published to date that still employ this concept. The first drawback of even time discretization is that it inherently results in a very large number of binary variables, particularly when the granularity of the problem is too small compared to the time horizon of interest. The second drawback is that accurate representation of time might necessitate even smaller time intervals with more binary variables. Even discretization of time is depicted in Fig. 1.8a.

Recent approaches tend to adopt the uneven discretization of the time horizon of interest wherein each time point along the time horizon coincides with either the start or the end of a task (Schilling and Pantelides, 1996). In addition to accurate representation of time this approach results in much smaller number of time points, hence fewer binary variables, as shown in Fig. 1.8b.

1.5 Recipe Representations

Another important aspect of batch plants relates to the representation of the recipe which is invariably the underlying feature of the resultant mathematical formulation. The most common representation of recipe in the published literature is the state task network (STN) that was proposed by Kondili et al. (1993), which comprises of 2 types of nodes, viz. state nodes and task nodes. The state nodes represent all the materials that are processed within the plant. These are broadly categorized into feeds, intermediates and final products. On the other hand, task nodes represent unit operations or tasks that are conducted in various equipment units within the process.

Very similar to the STN is the state sequence network (SSN) that was proposed by Majozi and Zhu (2001). The fundamental, and perhaps subtle, distinction between the SSN and the STN is that the tasks are not explicitly declared in the SSN, but indirectly inferred by the changes in states. A change from one state to another, which is simply represented by an arc, implies the existence of a task. Consequently, the mathematical formulation that is founded on this recipe representation involves only states and not tasks. The strength of the SSN lies in its ability to utilize information pertaining to tasks and even the capacity of the units in which the tasks are conducted by simply tracking the flow of states within the network. Since this representation and its concomitant mathematical formulation constitute the cornerstone of this textbook, it is presented in detail in the next chapter.

To illustrate the difference between SSN and STN, consider a simple batch process flowsheet shown in Fig. 1.9 (Ierapetritou and Floudas, 1998). The process involves 3 consecutive tasks, i.e. reaction, mixing and purification, and 4 consecutive states. The corresponding STN for this flowsheet is shown in Fig. 1.10a, whilst the SSN is shown in Fig. 1.10b. The existence of the arc between state 1 and state 2 signifies the presence of a task, which corresponds to mixing in this particular case.

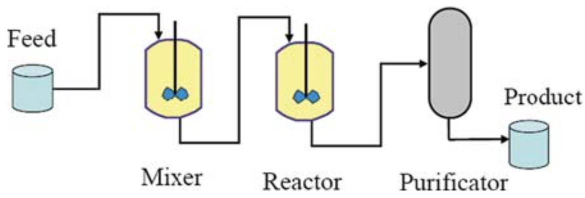


Fig. 1.9 Illustrative example (Ierapetritou and Floudas, 1998)

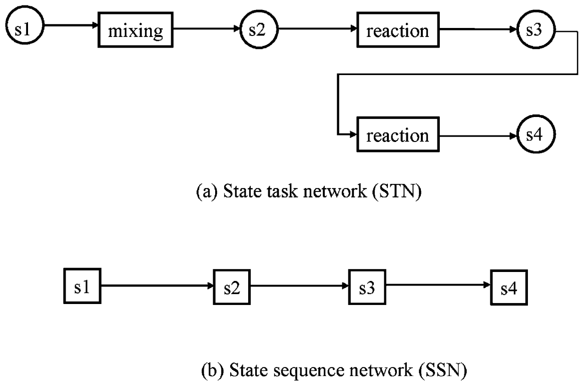


Fig. 1.10 (**a**) State task network vs (**b**) state sequence network for the illustrative example

The other popular recipe representations are the resource task network (RTN) proposed by Schilling and Pantelides (1996) and the schedule graph or S-graph (Sanmartí et al., 1998). The RTN is an enhanced version of the STN with resources representing feeds, intermediates, products, energy, manpower, storage and transportation facilities and the tasks as defined for the STN, but also including transportation, cleaning and storage. The S-graph on the other hand is a graph theoretic framework with a unique capability of exploiting the structure of the problem with substantial computational benefits. This framework, however, has been applied mainly in batch plant scheduling problems without FIS operational policy.

1.6 Batch Chemical Process Integration: Why Is It Necessary?

The foregoing sections give a fair background to the nature and intricate features of batch chemical plants. This background information is intended to cultivate an appreciation for the extent of distinction between batch and continuous chemical plants. It is expected that the reader is at this stage convinced that the depth of the chasm between the two types of processes cannot be simply bridged by adapting methodologies founded on continuous processes to batch processes without significantly losing accuracy.

Process integration, as a concept, has been applied almost exclusively to continuous processes for more than 3 decades. Most of the process integration methodologies tend to be of graphical nature, which allows the designer to be part of the optimization process. Typical outstanding examples include Pinch Analysis (Umeda et al., 1979; Linnhoff et al., 1979), which was developed for energy optimization and heat exchanger network (HEN) synthesis, Mass Exchanger Network (MEN) analysis, synthesis and optimization (El-Halwagi and Manousiouthakis, 1989; 1990a, b) and WaterPinch (Wang and Smith, 1994) that was developed for wastewater minimization in continuous plants at steady state. All these methodologies cannot be readily applied to batch processes since they appropriately exclude the time dimension which is overridden at steady state.

The intrinsic time dimension in batch chemical processes, as highlighted in Section 1.4, makes it rather difficult to confine the analysis of these processes to 2 dimensions for proper graphical interpretation. Consequently, most of the practical techniques in addressing batch plants tend to be of mathematical nature. The significant advance of mathematical tools and computational power suggest that this trend will remain true well into the future. However, the reluctance of batch chemical processes to be reduced to 2 dimensions is not the main reason for their exclusion from mainstream process integration analysis. Generally, it is considered true that batch processes are not as energy intensive as their continuous counterparts. Furthermore, it could be argued if the quantities of freshwater use and wastewater generation in batch processes warrant the need for dedicated effort in developing specific methodologies. However, the validity of both these perceptions has been significantly challenged with increased understanding of batch chemical plants.

Firstly, it is now understood that some of the batch operations, like dairy and brewing, tend to be as energy intensive as continuous processes (Mignon and Hermia, 1993). Secondly, the nature of most batch chemical plants is such that the effluent is of very high toxicity, albeit in small amounts. Typical examples are the agrochemical and pharmaceutical industries. Indeed, these 2 observations warrant the need for the dedicated effort in developing methodologies that are specific to batch chemical plants. All the material presented in the forthcoming chapters of this textbook is a contribution towards achieving this goal.

References

El-Halwagi, M.M., Manousiouthakis, V., 1989. Synthesis of mass exchange networks. AIChE J., 35(8): 1233–1244

El-Halwagi, M.M., Manousiouthakis, V., 1990a. Simultaneous synthesis of mass-exchange and regeneration networks. AIChE J., 36(8): 1209–1219

El-Halwagi, M.M., Manousiouthakis, V., 1990b. Automatic synthesis of mass-exchange networks with single-component targets. Chem. Eng. Sci., 45(9): 2813–2831

Ierapetritou, M.G., Floudas, C.A., 1998. Effective continuous-time formulation for short-term scheduling. 1. Multipurpose batch processes. Ind. Eng. Chem. Res., 37: 4341–4359

Kondili, E., Pantelides, C.C., Sargent, R.W.H., 1993. A general algorithm for short-term scheduling of batch operations-I. MILP formulation. Comp. Chem. Eng., 17(2): 211–227

Linhoff, B., Mason, D.R., Wardle, I., 1979. Understanding heat exchanger networks. Comput. Chem. Eng., 3: 295–302

Majozi, T., Zhu, X.X., 2001. A novel continuous time MILP formulation for multipurpose batch plants. 1. Short-term scheduling. Ind. Eng. Chem. Res., 40(25): 5935–5949
Mignon, D., Hermia, J., 1993. Using BATCHES for modeling and optimizing the brewhouses of an industrial brewery. Comput. Chem. Eng., 17(Suppl.): S51–S56.
Sanmartí, E., Friedler, F., Puigjaner, L., 1998. Combinatorial technique for short term scheduling of multipurpose batch plants based on schedule-graph representation. Comput. Chem. Eng., 22(Suppl.): S847–S850
Schilling, G., Pantelides, C.C., 1996. A simple continuous-time process scheduling formulation and a novel solution algorithm. Comput. Chem. Eng., 20(Suppl.): S1221–1226
Umeda, T., Harada, T., Shiroko, K., 1979. A thermodynamic approach to the synthesis of heat integration systems in chemical processes. Comput. Chem. Eng., 3: 273–282
Wang, Y.P., Smith, R., 1994. Wastewater minimization. Chem. Eng. Sci., 49(7): 981–1002

Chapter 2
Short-Term Scheduling

Overview In this chapter, the concept of the state sequence network (SSN) representation, which is the cornerstone of the material entailed in this textbook, is presented. This representation is based on states only, eliminating the use of tasks and units. Using this new representation as a basis, a novel continuous time MILP formulation for short-term scheduling of multipurpose batch plants is developed (Majozi and Zhu, 2001, Ind. Eng. Chem. Res., 40(25): 5935–5949). The presented formulation can readily be extended to intermediate due date scenarios. Time points are used to denote the use or production of a particular state. This formulation leads to a small number of binary variables and much better results when compared to other continuous time formulations published in literature. The reduced number of binary variables is a result of considering states only, thereby eliminating binary variables corresponding to tasks and units. This method has been applied to literature examples and industrial problems which show significant improvement in reducing the number of binary variables, hence CPU times. The last section of this chapter introduces the concept of units aggregation in reducing the binary dimension of large-scale problems. This makes it possible for the method to solve large-scale industrial problems. In the forthcoming chapters of this textbook it will be demonstrated how this formulation is applied in the context of batch process integration with the ultimate aim of freshwater and energy optimization.

2.1 Problem Statement

The scheduling problem that is considered in this chapter can be stated as follows.
Given:

(i) the production recipe for each product, including mean processing times in each unit operation,
(ii) the available units and their capacities,
(iii) the maximum storage capacity for each material and
(iv) the time horizon of interest.

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_2,

determine:

(i) the optimal schedule for tasks within the time horizon of interest,
(ii) the amount of material processed in each unit at any particular point in time within the time horizon and
(iii) the amount delivered to customers over the entire time horizon.

2.2 State Sequence Network

Figure 2.1 illustrates the building blocks of the state sequence network (SSN). Figure 2.1a shows the transition from state s to state s'. For an example, this could be a washing operation with s as washing water and s' as effluent. This implies that there has to be a unit operation between these two due to the change in states. Figure 2.1b shows the mixing of different states to yield a new state, e.g. raw materials entering a reactor to yield a product. Figure 2.1c illustrates a splitting/separation unit with state s as the input and states s' and s'' as outputs.

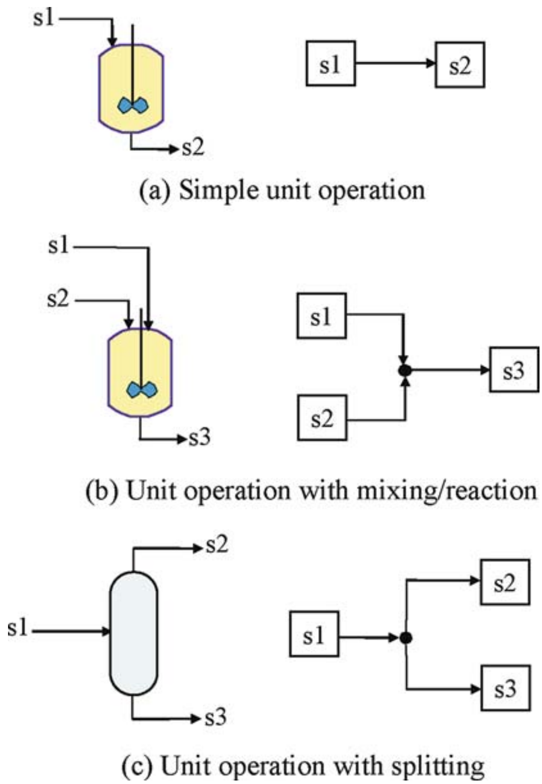


Fig. 2.1 Building blocks for the state sequence network

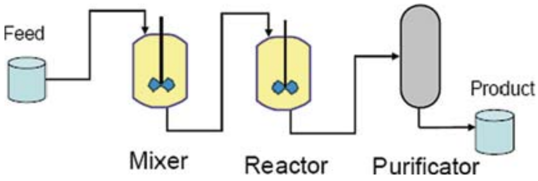


Fig. 2.2 Plant flowsheet for the literature example

In order to facilitate understanding of this concept, a literature problem (Ierapetritou and Floudas, 1998) is presented in Fig. 2.2.

For comparison, the STN and SSN corresponding to this example are shown in Fig. 2.3a, b, respectively.

The SSN for this example takes its simplest form, since there is no mixing or splitting of independent states. In the later examples, it will be demonstrated how the SSN is constructed if there is mixing or splitting of states. In the SSN representation, only states are considered while tasks and units are implicitly incorporated. This representation is developed by realising that (i) the capacity of a unit in which a particular state is used or produced sets an upper limit on the amount of state used or produced by the corresponding task; (ii) the presence of a particular state in an operation corresponds to the existence of a corresponding task; and (iii) the usage of state s (input) corresponds to the production of state s' (output). Time points, as introduced by Schilling and Pantelides (1996) are used in this formulation. This is similar to the event point concept used by Ierapetritou and Floudas (1998), except that there are fundamental differences in the formulation of sequence and duration constraints. Figure 2.4 shows how the time points are defined in the time horizon of interest.

In their formulation, Ierapetritou and Floudas (1998), separated task and unit events by assigning corresponding binary variables to tasks, $wv\,(i,p)$, and units, $yv\,(i,p)$, respectively. This led to an overall number of binary variables of $P(N_i+N_j)$, where P is the number of time points, whilst N_i and N_j are the numbers of tasks

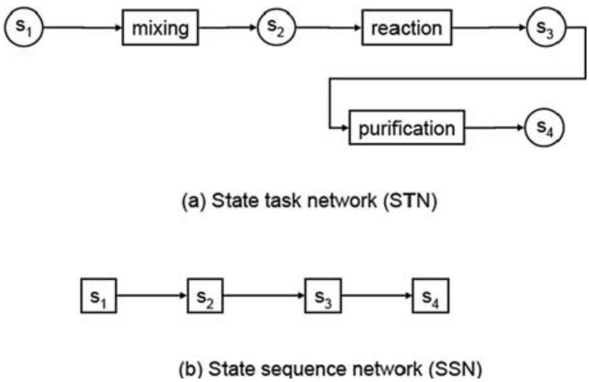


Fig. 2.3 (**a**) STN and (**b**) SSN presentations for the literature example

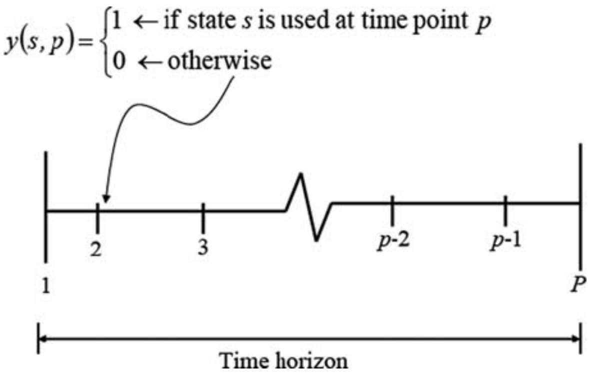


Fig. 2.4 Description of time points

and units, respectively. As a result, the example in Fig. 2.2 gave P (3+3) binary variables, since it consists of 3 tasks and 3 units. Exploiting the one-to-one correspondence between tasks and units further reduced this number of binary variables to $3 \times P$. This is readily achievable for processes in which only one task is performed in one unit. If a unit can perform more than one task as in the examples given later in this chapter, this reduction of binary variables is not straightforward. However, in the SSN based formulation the only binary variables involved are those corresponding to states, i.e. $y(s,p)$. It should also be noted that the usage of state s at time point p corresponds to the production of state s' at time point $p+1$, where s and s' are input and output states of a particular operation, respectively. In the event where many states are used simultaneously, which is frequently encountered in batch reactors and blending operations, only one state is assigned the binary variable $y(s,p)$ as all these states are used at the same time point. The state that has been assigned the binary variable is termed an effective state. Therefore, the resulting number of binary variables from this formulation is $E \times P$, where E is the number of effective states involved in the process, and $P$is the total number of time points used in the formulation. Worth the emphasis, is the fact that the choice of effective states is solely at the discretion of the user. However, once the choice of effective states has been made, it should remain consistent throughout the formulation.

In the example of Fig. 2.2, the formulation that follows results in $3 \times P$ binary variables. This is due to the fact that states $s1$, $s2$ and $s3$ are the effective states. However, in using aggregation models that are only applicable to in-phase operation, as presented in the last section of the chapter, the resulting number of binary variables is always $S \times P$, where S is the number of processing stages involved in the process. Each processing stage can have more than one processing units (processors). It should, therefore, be noticed that the aggregation model is the simplification of the general model using the effective states.

All the results presented in this chapter were obtained using GAMS 2.5/OSL in a 600 MHz Pentium III processor, unless otherwise stated.

2.3 Mathematical Model

The mathematical model presented in this chapter comprises of the following sets and parameters. In cases where a variable or a parameter has not been declared and defined in the list, it is described accordingly in the text.

Sets

$$P = \{p \,|p = \text{time point}\}$$

$$J = \{j \,|\, j = \text{unit}\}$$

$$S_{\text{in}} = \{S_{\text{in}} \,|S_{\text{in}} = \text{input state into any unit}\}$$

$$S_{\text{out}} = \{S_{\text{out}} \,|S_{\text{out}} = \text{output state from any unit}\}$$

$$S = \{s \,|s = \text{anystate}\} = S_{\text{in}} \cup S_{\text{out}}$$

$$S_{\text{in},j} = \left\{S_{\text{in},j} \,\middle|S_{\text{in},j} = \text{input state into unit } j\right\} \subseteq S_{\text{in}}$$

$$S^*_{\text{in},j} = \left\{S^*_{\text{in},j} \,\middle|S^*_{\text{in},j} = \text{effective state into unit } j\right\} \subseteq S_{\text{in},j}$$

$$S_{\text{out},j} = \left\{S_{\text{out},j} \,\middle|S_{\text{out},j} = \text{out state from unit } j\right\} \subseteq S_{\text{out}}$$

Variables

$t_p\left(s_{\text{out},j}, p\right)$	time at which a state is produced from unit j at time point p
$t_u\left(s_{\text{in},j}, p\right)$	time at which a state is used in or enters unit j at time point p
$q_s\,(s,p)$	amount of state s stored at time point p
$m_p\left(s_{\text{out},j}, p\right)$	amount of state produced from unit j at time point p
$m_u\left(s_{\text{in},j}, p\right)$	amount of state used in or enters unit j at time point p
$y\left(s^*_{\text{in},j}, p\right)$	binary variable associated with usage of state s at time point p
$d\,(s_{\text{out}}, p)$	amount of state delivered to customers at time point p

Parameters

$a\left(s^*_{\text{in},j}\right)$	duration parameter associated with variable batch size
$b\left(s^*_{\text{in},j}\right)$	duration parameter associated with variable batch size

$t^L\left(s^*_{\text{in},j}\right)$	minimum processing time in unit j corresponding to a particular task
$t^U\left(s^*_{\text{in},j}\right)$	maximum processing time in unit j corresponding to a particular task
V^U_j	maximum design capacity of a particular unit j
V^L_j	minimum design capacity of a particular unit j
H	time horizon of interest
$\tau\left(s^*_{\text{in},j}\right)$	mean processing time for a state
$Q^0_s(s)$	initial amount of state s stored
$Q^U_s(s)$	maximum amount of state s stored within the time horizon of interest
$CP(s)$	Selling price of product s, $s =$ product

Capacity Constraints

$$V^L_j y\left(s^*_{\text{in},j},p\right) \le \sum_{s_{\text{in},j}} m_u\left(s_{\text{in},j},p\right) \le V^U_j y\left(s^*_{\text{in},j},p\right), \forall j \in J, p \in P, \forall s_{\text{in},j} \in S_{\text{in},j} \tag{2.1}$$

This constraint implies that the total amount of all the states consumed at time point p is limited by the capacity of the unit which consumes the states. The U and L superscripts denote the upper and the lower bound on capacity. According to constraint (2.1), states will be consumed in a particular unit $j$if the corresponding effective state is used at time point p.

Material Balances

$$\sum_{s_{\text{in},j}} m_u\left(s_{\text{in},j},p-1\right) = \sum_{s_{\text{out},j}} m_p\left(s_{\text{out},j},p\right), \forall p \in P, p > p_1, j \in J, \quad \forall s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j} \tag{2.2}$$

$$q_s(s,p_1) = Q^0_s(s) - m_u(s,p_1), s \ne \text{product}, \forall s \in S \tag{2.3}$$

$$q_s(s,p) = q_s(s,p-1) - m_u(s,p), s = \text{feed}, \forall s \in S, \forall p \in P, p > p_1 \tag{2.4}$$

$$q_s(s,p) = q_s(s,p-1) + m_p(s,p) - m_u(s,p), s \ne \text{product, feed}, \forall s \in S \quad \forall p \in P, p > p_1 \tag{2.5}$$

$$q_s(s,p_1) = Q^0_s(s) - d(s,p_1), s = \text{product}, \forall s \in S \tag{2.6}$$

$$q_s(s,p) = q(s,p-1) + m_p(s,p) - d(s,p), s = \text{product, byproduct}, \forall s \in S \quad \forall p \in P, p > p_1 \tag{2.7}$$

Constraint (2.2) is the material balance around a particular unit j. It implies that the sum of the masses for all the input states used at time point p –1 should be equal to the sum of the masses for all the output states produced at time point p. Constraint (2.3) states that the amount of state s stored at the first time point, is the difference between the amount stored before the beginning of the process and that being utilised at the first time point. Constraint (2.4) only applies to the feed, since it is the state that is only used in the process. Constraint (2.5) only applies to intermediates, since they are both produced and used in the process. Constraints (2.6) and (2.7) only apply to products and byproducts, since they are the only states that have to be taken out of the process as shown by the terms $d(s, p)$.

Duration Constraints (Batch Time as a Function of Variable Batch Size)

In this section the duration constraints are modelled as a function of batch size. The following constraints show how this effect is modelled in the proposed approach using the SSN representation.

$$t_p\left(s_{\text{out},j},p\right)=t_u\left(s^*_{\text{in},j},p-1\right)+a\left(s^*_{\text{in},j}\right)y\left(s^*_{\text{in},j},p-1\right)+b\left(s^*_{\text{in},j}\right)\sum_{s_{\text{in},j}}m_u\left(s,p-1\right),$$
$$\forall j\in J,\forall p\in P,p>p_1,\forall s_{\text{in},j}\in S_{\text{in},j},s_{\text{out},j}\in S_{\text{out},j} \tag{2.8}$$

$$b\left(s^*_{\text{in},j}\right)=\frac{t^U\left(s^*_{\text{in},j}\right)-t^L\left(s^*_{\text{in},j}\right)}{V_j^U-V_j^L},\forall j\in J,\forall s^*_{\text{in},j}\in S_{\text{in},j} \tag{2.9}$$

$$a\left(s^*_{\text{in},j}\right)=\tau\left(s^*_{\text{in},j}\right)\left(1-\upsilon\left(s^*_{\text{in},j}\right)\right)=t^L\left(s^*_{\text{in},j}\right),\forall j\in J,\forall s^*_{\text{in},j}\in S_{\text{in},j} \tag{2.10}$$

$$t^U\left(s^*_{\text{in},j}\right)=\tau\left(s^*_{\text{in},j}\right)\left(1+\upsilon\left(s^*_{\text{in},j}\right)\right),\forall j\in J,\forall s^*_{\text{in},j}\in S_{\text{in},j} \tag{2.11}$$

The parameter for variable batch time is defined by constraint (2.9). This gives the amount of time required to process a unit amount of a batch corresponding to a particular effective state in a corresponding unit operation. Constraint (2.10) denotes the minimum processing time for the effective state in the corresponding unit operation. This is, in essence, the minimum residence time of a batch within a unit operation. In constraints (2.10) and (2.11), $\upsilon\left(s^*_{\text{in},j}\right)$ is the percentage variation in processing time based on operational experience.

Duration Constraints (Batch Time Independent of Batch Size)

In a situation where duration is constant regardless of the batch size, the duration constraint assumes the following form.

$$t_p\left(s_{\text{out},j},p\right) = t_u\left(s^*_{\text{in},j},p-1\right) + \tau\left(s^*_{\text{in},j}\right) y\left(s^*_{\text{in},j},p-1\right), \forall p \in P, p > p_1, \\ \forall\, s_{\text{out},j} \in S_{\text{out},j}, s^*_{\text{in},j} \in S_{\text{in},j} \tag{2.12}$$

Sequence Constraints

$$t_u\left(s_{\text{in},j},p\right) \geq \sum_{s_{\text{in},j}} \sum_{s_{\text{out},j}} \sum_{p' \leq p} \left(t_p\left(s_{\text{out},j},p'\right) - t_u\left(s_{\text{in},j},p'-1\right)\right), \\ \forall j \in J, p' \in P, p > p_1, \forall s_{\text{out},j} \in S_{\text{out},j}, s_{\text{in},j} \in S_{\text{in},j} \tag{2.13}$$

$$t_u\left(s_{\text{in},j},p\right) \geq t_p\left(s_{\text{out},j},p'\right) - H\left(2 - y\left(s^*_{\text{in},j},p\right) - y\left(s^*_{\text{in},j},p'-1\right)\right), \\ \forall j \in J, p, p' \in P, p' > p_1, p' \leq p, \forall s_{\text{out},j} \in S_{\text{out},j}, s_{\text{in},j} \in S_{\text{in},j}, s^*_{\text{in},j} \to s_{\text{out},j} \tag{2.14}$$

$$t_u\left(s_{\text{in},j},p\right) \geq t_p\left(s_{\text{out},j'},p'\right) - H\left(2 - y\left(s^*_{\text{in},j},p\right) - y\left(s^*_{\text{in},j'},p'-1\right)\right), \\ \forall j \in J, p, p' \in P, p' > p_1, p' \leq p, \forall s_{\text{in},j} \in S_{\text{in},j}, s_{\text{in},j} = s_{\text{out},j'}, s^*_{\text{in},j'} \to s_{\text{out},j'} \tag{2.15}$$

Constraints (2.13) and (2.14) imply that state s can only be used in a particular unit, at any time point, after all the previous states have been processed. Constraint (2.13) is only relevant in situations where more than one task can be conducted in one unit, otherwise it is redundant in the presence of constraints (2.14) and (2.15). Constraint (2.15) stipulates that a state can only be processed at a particular time point p in a particular unit j after it has been produced from another unit j'. In case of a recycle, j is the same as j'. It is worthy of note that constraints (2.14) and (2.15) are only applicable to intermediates, since they are the only states that are both produced and used. For clarity of notation which is adopted throughout this textbook, $s \to s'$ implies that state s' is formed from a task that uses state s as its input.

Assignment Constraint

The assignment constraint is aimed at ensuring that only one task is conducted in a unit at any time point. It is, therefore, apparent that the assignment constraint is only necessary if more than one task can be performed in a given unit. Otherwise, it is also redundant.

$$\sum_{s^*_{\text{in},j}} y\left(s^*_{\text{in},j},p\right) \leq 1, \forall p \in P, j \in J, s^*_{\text{in},j} \in S_{\text{in},j} \tag{2.16}$$

Time Horizon Constraints

$$t_u\left(s_{\text{in},j},p\right) \leq H, \forall s_{\text{in},j} \in S_{\text{in},j}, p \in P, j \in J \tag{2.17}$$

$$t_u\left(s_{\text{out},j},p\right) \leq H, \forall s_{\text{out},j} \in S_{\text{out},j}, p \in P, j \in J \tag{2.18}$$

Constraints (2.17) and (2.18) respectively stipulate that the usage or production of state should be within the time horizon of interest.

Storage Constraints

$$q_s\left(s,p\right) \leq Q^U\left(s\right), \forall s \in S, p \in P \tag{2.19}$$

Constraint (2.19) states that the amount of state s stored at each time point cannot exceed the maximum allowed.

Objective Function

The objective function for this formulation is the maximisation of product throughput or revenue.

$$\text{Maximize} \sum_s \sum_p CP\left(s\right) d\left(s,p\right), s = \text{product}, s \subset S, p \in P \tag{2.20}$$

2.4 Literature Examples

2.4.1 First Literature Example

In this section, the above mathematical model is applied to a literature example shown in Fig. 2.2 (Ierapetritou and Floudas, 1998). The SSN representation is given in Fig. 2.3b. Table 2.1 gives data for this example. 5 time points and a 12-h time horizon were used. Using less time points leads to a suboptimal solution with an objective value of 50, and using more time points than 5 did not improve the solution. It is worthy of note that, in this particular example, constraint (2.13) is redundant as mentioned earlier, since each unit is only performing one task.

Capacity Constraints

State $s1$
$m_u\left(s1,p\right) \leq 100y\left(s1,p\right), \forall p \in P$

State $s2$
$m_u\left(s2,p\right) \leq 75y\left(s2,p\right), \forall p \in P$

State $s3$
$m_u\left(s3,p\right) \leq 50y\left(s3,p\right), \forall p \in P$

Table 2.1 Data for the literature example (Ierapetritou and Floudas, 1998)

Unit	Capacity	Suitability	Mean processing time (τ)
Unit 1	100	Mixing	4.5
Unit 2	75	Reaction	3.0
Unit 3	50	Purification	1.5
State	Storage capacity	Initial amount	Price
State 1	Unlimited	Unlimited	0.0
State 2	100	0.0	0.0
State 3	100	0.0	0.0
State 4	Unlimited	0.0	1.0

Material Balances

Unit mass balances – mixer

$$m_p(s2,p) = m_u(s1,p-1), \forall p \in P, p > p_1$$

Unit mass balances – reactor

$$m_p(s3,p) = m_u(s2,p-1), \forall p \in P, p > p_1$$

Unit mass balances – purificator

$$m_p(s4,p) = m_u(s3,p-1), \forall p \in P, p > p_1$$

State $s1$

$$q_s(s1,p_1) = Q_s^0(s1) - m_u(s1,p_1)$$
$$q_s(s1,p) = q_s(s1,p-1) - m_u(s1,p), \forall p \in P,\ p > p_1$$

State $s2$

$$q_s(s2,p_1) = Q_s^0(s2) - m_u(s2,p_1)$$
$$q_s(s2,p) = q_s(s2,p-1) + m_p(s2,p) - m_u(s2,p), \forall p \in P,\ p > p_1$$

State $s3$

$$q_s(s3,p_1) = Q_s^0(s3) - m_u(s3,p_1)$$
$$q_s(s3,p) = q_s(s3,p-1) + m_p(s3,p) - m_u(s3,p), \forall p \in P,\ p > p_1$$

State $s4$

$$q_s(s4,p_1) = Q_s^0(s4) - d(s4,p_1)$$
$$q_s(s4,p) = q_s(s4,p-1) + m_p(s4,p) - d(s4,p), \forall p \in P,\ p > p_1$$

Duration Constraints (Batch Time as a Function of Batch Size)

$$t_p(s2,p) = t_u(s1,p-1) + 3y(s1,p-1) + 0.03m_u(s1,p-1), \forall p \in P, p > p_1$$
$$t_p(s3,p) = t_u(s2,p-1) + 2y(s2,p-1) + 0.0267m_u(s2,p-1), \forall p \in P, p > p_1$$
$$t_p(s4,p) = t_u(s3,p-1) + y(s3,p-1) + 0.02m_u(s3,p-1), \forall p \in P, p > p_1$$

The coefficients for $m_u(s,p)$ and $y(s,p)$ are defined by constraints (2.9) and (2.10), respectively.

Sequence Constraints

Since each unit can only perform one task, constraint (2.13) is redundant.

States $s1$ *and* $s2$

$$t_u(s1,p) \geq t_p(s2,p) - H(2 - y(s1,p) - y(s1,p-1)), \forall p \in P, p > 1$$

States $s2$ *and* $s3$

$$t_u(s2,p) \geq t_p(s3,p') - H(2 - y(s2,p) - y(s2,p'-1)), \forall p \in P, p' \leq p, p' > 1$$

$$t_u(s2,p) \geq t_p(s2,p') - H(2 - y(s2,p) - y(s1,p'-1)), \forall p \in P, p' \leq p, p' > 1$$

States $s3$ *and* $s4$

$$t_u(s3,p) \geq t_p(s4,p') - H(2 - y(s3,p) - y(s3,p'-1)), \forall p \in P, p' \leq p, p' > 1$$

$$t_u(s3,p) \geq t_p(s3,p') - H(2 - y(s3,p) - y(s2,p'-1)), \forall p \in P, p' \leq p, p' > 1$$

These correspond to constraints (2.14) and (2.15) given in the mathematical model.

Time Horizon Constraints

State $s1$

$$t_u(s1,p) \leq 12, \forall p \in P$$

State $s2$

$$t_p(s2,p) \leq 12, t_u(s2,p) \leq 12, \forall p \in p$$

State $s3$

$$t_p(s3,p) \leq 12, t_u(s3,p) \leq 12, \forall p \in p$$

State $s4$

$$t_p(s4,p) \leq 12, \forall p \in p$$

Storage Constraints

$$q_s(s2,p) \leq 100, \forall p \in p$$

$$q_s(s3,p) \leq 100, \forall p \in p$$

Objective Function

$$\text{Maximize}\, R = \sum_p d(s4,p), \forall p \in P$$

In this formulation, the only binary variables involved are $y(s1,p)$,$y(s2,p)$ and $y(s3,p)$ corresponding to states $s1$, $s2$ and $s3$, respectively. Therefore the total number of binary variables is $3 \times P$.

Computational Results

Five time points and a 12-h time horizon were used for this example. The results from this proposed method and from the methods proposed by Ierapetritou and Floudas (1998), Zhang (1995), and Schilling and Pantelides (1996) are shown in Table 2.2.

The results in the second and third columns were obtained using GAMS 2.5/OSL in a 600 MHz Pentium III processor, while those in the fourth and fifth columns were taken directly from Ierapetritou and Floudas (1998). The approach based on the SSN representation gives an objective value of 71.473 and requires only 15 binary variables, compared to 48 and 46 binary variables required in approaches proposed by Zhang, and Schilling and Pantelides, respectively. The formulation by Ierapetritou and Floudas (1998) initially consisted of 30 binary variables that were later reduced to 15 by exploiting one to one correspondence of units and tasks. It

Table 2.2 Results for the first literature example

	This approach	Ierapetritou and Floudas	Zhang	Schilling and Pantelides
NTP	5	5	7	6
NC	158	177	263	220
NV	103	101	187	157
NB	15	30 (15)	48	46
MILP solution	71.473	71.518	71.45	71.47
Relaxed objective	100	100	149.99	170.79
CPU time (s)	0.168	0.219	21.9	N/A

NTP = number of time points; NC = number of constraints; NV = total number of variables; NB = number of binary variables

Table 2.3 Values of the binary variables at different time points for the literature example

$y(s, p)$	p_1	p_2	p_3	p_4	p_5
$s1$	1	1	0	0	0
$s2$	0	1	1	0	0
$s3$	0	0	1	1	0

is also worthy of note that the proposed formulation requires the least number of constraints, i.e. 158, and shortest CPU time (0.168 s). Using four time points led to an objective value of 50 and using more time points did not improve the objective function.

Table 2.3 shows the values of the binary variables at different time points, which forms the basis for the construction of the Gantt chart shown in Fig. 2.5. Since the last time point, i.e. p_5, corresponds to the end of the time horizon, all the binary variables are zero as no state can be used. The value of a binary variable signifies the usage of a particular state at a particular time point. This corresponds to the beginning of a particular task in a particular unit. As an illustration, the value of the binary variable corresponding to the usage of state $s1$ at the first time point is unity (Table 2.3), which implies that mixing commences in the mixer at this time point. According to the Gantt chart corresponding to Table 2.3 as shown in Fig. 2.5, this particular task begins at 0 h and finishes at 4.665 h. It is, therefore, apparent that both the unit and the task are described by a single binary variable y (s, p), which is the essence of the SSN approach. According to Table 2.3, this task (mixing) also begins at the second time point. This corresponds to mixing that begins in the mixer at 4.665 h and finishes at 8.144 h as shown in Fig. 2.5. The other time points that correspond to states $s2$ and $s3$, which respectively represent reaction and purification, are also interpreted similarly.

The values shown above the horizontal bars in the Gantt chart represent the quantity that is processed in a unit. In accordance with the formulation, Fig. 2.5 shows that the smaller the batch, the shorter the processing time in the same unit. The

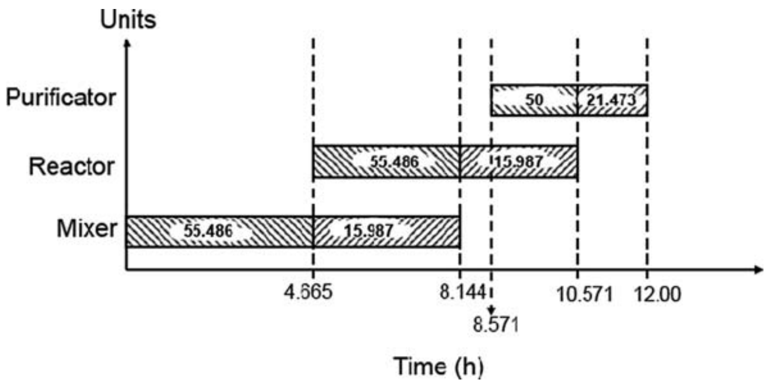


Fig. 2.5 Gantt chart for the literature example

objective value of 71.518 reported by Ierapetritou and Floudas (1998) was due to rounding off the parameter for batch size variation. The same value is obtained in this formulation if a similar rounding off is used.

2.4.2 Second Literature Example

The flowsheet for the second literature example is shown in Fig. 2.6 and the STN and SSN are given in Figs. 2.7a, b, respectively (Ierapetritou and Floudas, 1998).

Choice of Effective States

Since this problem involves more than one state entering some units, i.e. reactors 1 and 2, it is necessary to choose effective states before proceeding. It should be re-emphasised that the only requirement for the choice of the effective states is that, only one of the input states that are used simultaneously in a particular unit should be chosen. Therefore, any given set of effective states is not unique. However, once the choice of effective states has been made it should remain consistent throughout the formulation. Moreover, the fact that the choice of effective states is not unique does not change the binary dimension of the problem. As an illustration, the following three are possible sets of effective states. Each state in these sets corresponds to a particular task. The first to the fifth elements in each of the sets correspond to heating, reaction 1, reaction 2, reaction 3 and separation, respectively. It is evident that there are 5 more possible sets of effective states corresponding to this example.

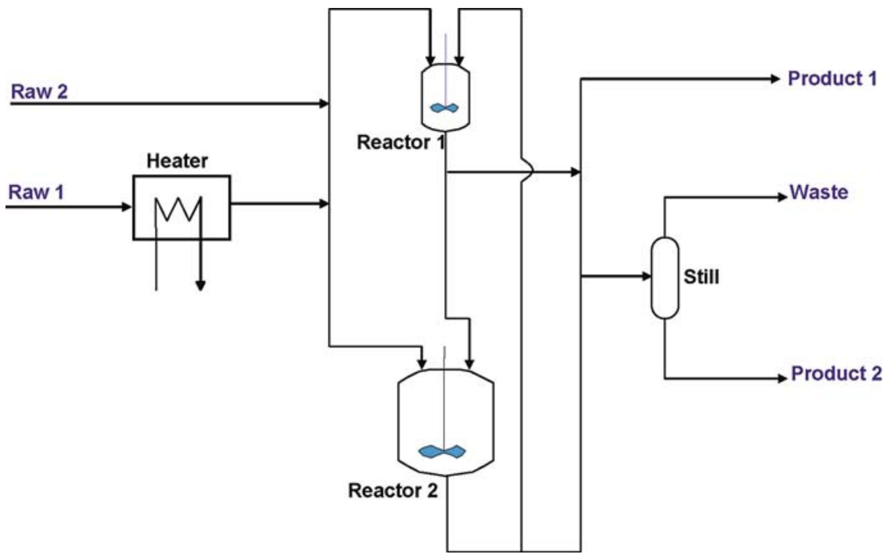


Fig. 2.6 Flowsheet for the second literature example

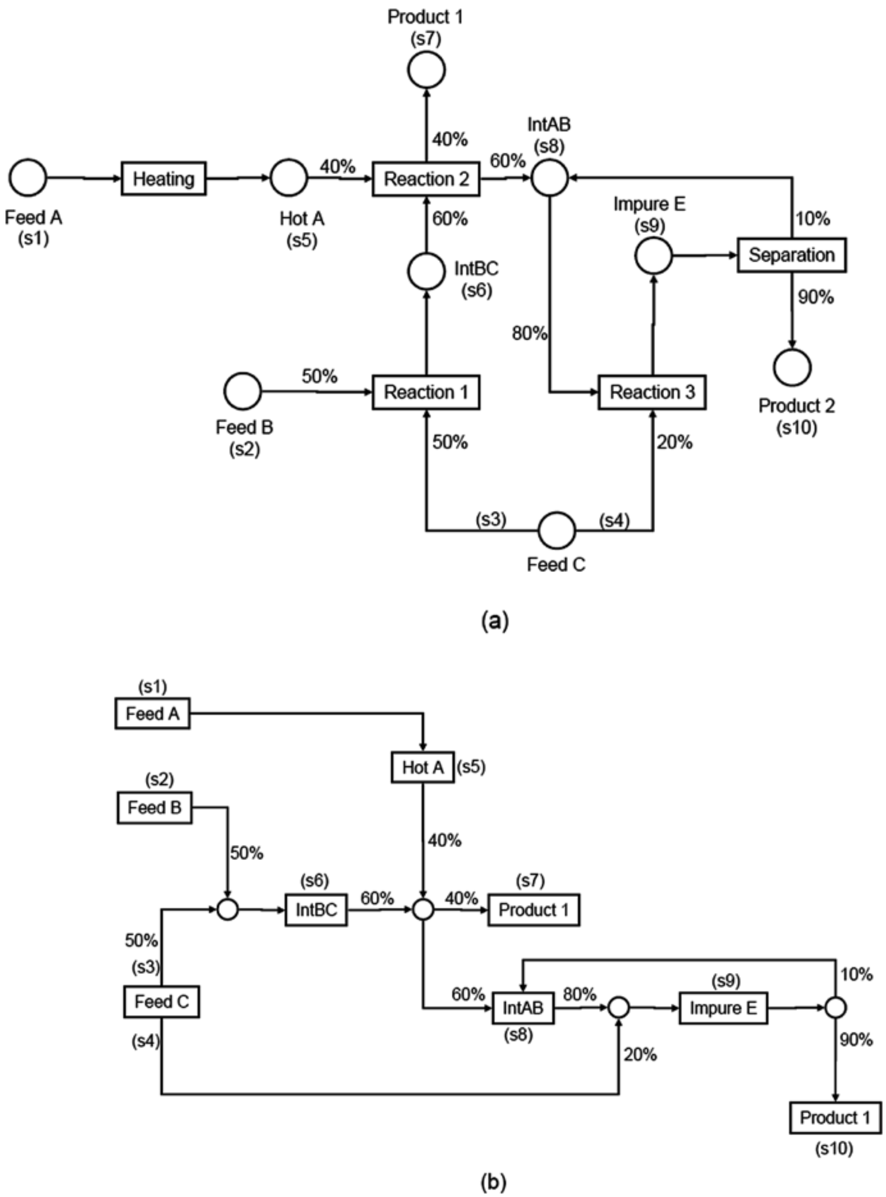


Fig. 2.7 (**a**) STN and (**b**) SSN for the second literature example

Table 2.4 Data for the second literature example

Unit	Capacity	Suitability	Mean processing time (τ)
Heater ($j = 1$)	100	Heating	1.0
Reactor 1 ($j = 2$)	50	Reaction 1, 2, 3	2.0, 2.0, 1.0
Reactor 2 ($j = 3$)	80	Reaction 1, 2, 3	2.0, 2.0, 1.0
Still ($j = 4$)	200	Separation	1 for product 2, 2 for IntAB
State	**Storage capacity**	**Initial amount**	**Price**
Feed A	Unlimited	Unlimited	0.0
Feed B	Unlimited	Unlimited	0.0
Feed C	Unlimited	Unlimited	0.0
Hot A	100	0.0	0.0
IntAB	200	0.0	0.0
IntBC	150	0.0	0.0
Impure E	200	0.0	0.0
Product 1	Unlimited	0.0	10.0
Product 2	Unlimited	0.0	10.0

$$S^*_{\text{in},j} = \left\{s1, s2_{\text{in},j}, s6_{\text{in},j}, s8_{\text{in},j}, s9\right\}, j = 2, 3$$
$$S^*_{\text{in},j} = \left\{s1, s3_{\text{in},j}, s5_{\text{in},j}, s4_{\text{in},j}, s9\right\}, j = 2, 3$$
$$S^*_{\text{in},j} = \left\{s1, s2_{\text{in},j}, s5_{\text{in},j}, s4_{\text{in},j}, s9\right\}, j = 2, 3$$

Since there are 8 effective states for the overall problem, the resulting number of binary variables is $8 \times P$. Note that this number of effective states emanates from the fact that each of states $s2$, $s6$ and $s8$ can be fed to any of the 2 reactors. The first set of effective states was chosen throughout the formulation.

The data for this example appears in Table 2.4.

In this example, there are two reactors in which reactions 1, 2 and 3 can be performed. Equal mean reaction times for the different reactions in each of the reactors imply similar performances for the reactors. The overall process consists of four units, i.e. heater, reactor 1, reactor 2 and separator. In order to handle the usage of feed C in two distinct reactions, i.e. reactions 1 and 3, different states were assigned to each of the streams of feed C, i.e. states $s3$ and $s4$, respectively. In this example, scheduling is performed over an 8-h time horizon. It should be noted that reactors 1 and 2 are suitable for performing reactions 1, 2 and 3, which implies that constraint (2.13) is crucial. Constraints that exhibit similar structure to those presented in example 1 are not repeated.

Capacity Constraints

Reaction 1

Reactor 1($j = 2$):

$m_u(s2_{\text{in},2}, p) + m_u(s3_{\text{in},2}, p) \leq 50y(s2_{\text{in},2}, p), \forall p \in P$

Reactor 2(j= 3):

$$m_u(s2_{\text{in},3},p) + m_u(s3_{\text{in},3},p) \leq 80y(s2_{\text{in},3},p), \forall p \in P$$

Reaction 2

Reactor 1(j= 2):

$$m_u(s6_{\text{in},2},p) + m_u(s5_{\text{in},2},p) \leq 50y(s6_{\text{in},2},p), \forall p \in P$$

Reactor 2(j= 3):

$$m_u(s6_{\text{in},3},p) + m_u(s5_{\text{in},3},p) \leq 80y(s6_{\text{in},3},p), \forall p \in P$$

Reaction 3

Reactor 1(j= 2):

$$m_u(s8_{\text{in},2},p) + m_u(s4_{\text{in},2},p) \leq 50y(s8_{\text{in},2},p), \forall p \in P$$

Reactor 2(j= 3):

$$m_u(s8_{\text{in},3},p) + m_u(s4_{\text{in},3},p) \leq 80y(s8_{\text{in},3},p), \forall p \in P$$

Material Balances

Unit mass balances – reaction 1

$$m_u(s2_{\text{in},j},p-1) + m_u(s3_{\text{in},j},p-1) = m_p(s6_{\text{out},j},p), \forall p \in P, p > p_1, j = 2,3$$

Unit mass balances – reaction 2

$$m_u(s6_{\text{in},j},p-1) + m_u(s5_{\text{in},j},p-1) = \frac{10}{4} m_p(s7_{\text{out},j},p), \forall p \in P, p > p_1, j = 2,3$$

The coefficient for $m_p(s7_{\text{out},j}, p)$ is derived from the stoichiometric values given in the SSN.

Unit mass balances – reaction 3

$$m_u(s8_{\text{in},j},p-1) + m_u(s4_{\text{in},j},p-1) = m_p(s9_{\text{out},j},p), \forall p \in P, p > p_1, j = 2,3$$

Unit mass balances – separation

$$m_u(s9_{\text{in},4},p-1) = \frac{10}{9} m_p(s10_{\text{out},4},p), \forall p \in P, p > p_1$$

Stoichiometric constraints

$$m_u(s3_{\text{in},j},p) = m_u(s2_{\text{in},j},p), \forall p \in P, j = 2,3$$

$$m_u(s6_{\text{in},j},p) = \frac{60}{40} m_u(s5_{\text{in},j},p), \forall p \in P, j = 2,3$$

$$m_u(s8_{\text{in},j},p) = \frac{80}{20} m_u(s4_{\text{in},j},p), \forall p \in P, j = 2,3$$

These stoichiometric constraints are derived from the data given in the SSN.

Sequence Constraints

Since reactors 1 and 2 can conduct reactions 1, 2 and 3, constraint (2.13) is necessary as mentioned earlier. However, this constraint is not necessary for the heater and separator. Following is the constraint corresponding to constraint (13) for this problem.

$$t_u\left(s^*_{\text{in},j},p\right) \geq \sum_{p'=p2}^{p}\begin{pmatrix} t_p\left(s6_{\text{out},j},p'\right) - t_u\left(s2_{\text{in},j},p'-1\right) + t_p\left(s8_{\text{out},j},p'\right) \\ -t_u\left(s6_{\text{in},j},p'-1\right) + t_p\left(s9_{\text{out},j},p'\right) - t_u\left(s8_{\text{in},j},p-1\right)\end{pmatrix}, \forall p \in P,$$
$$s^*_{\text{in},j} = s2_{\text{in},j}, s6_{\text{in},j}, s8_{\text{in},j}, j = 2,3$$

The following constraints correspond to constraints (2.14) and (2.15) in the mathematical model.

States $s2$, $s6$ *and* $s8$

$$t_u\left(s2_{\text{in},j},p\right) \geq t_p\left(s6_{\text{out},j},p'\right) - H\left(2 - y\left(s2_{\text{in},j},p\right) - y\left(s2_{\text{in},j},p'-1\right)\right), \quad \forall p,p' \in P, p' > 1, p' \leq p, j = 2,3$$
$$t_u\left(s2_{\text{in},j},p\right) \geq t_p\left(s8_{\text{out},j},p'\right) - H\left(2 - y\left(s2_{\text{in},j},p\right) - y\left(s6_{\text{in},j},p'-1\right)\right), \quad \forall p,p' \in P, p' > 1, p' \leq p, j = 2,3$$
$$t_u\left(s2_{\text{in},j},p\right) \geq t_p\left(s9_{\text{out},j},p'\right) - H\left(2 - y\left(s2_{\text{in},j},p\right) - y\left(s8_{\text{in},j},p'-1\right)\right), \quad \forall p,p' \in P, p' > 1, p' \leq p, j = 2,3$$
$$t_u\left(s6_{\text{in},j},p\right) \geq t_p\left(s6_{\text{out},j},p'\right) - H\left(2 - y\left(s6_{\text{in},j},p\right) - y\left(s2_{\text{in},j},p'-1\right)\right), \quad \forall p,p' \in P, p' > 1, p' \leq p, j = 2,3$$
$$t_u\left(s6_{\text{in},j},p\right) \geq t_p\left(s8_{\text{out},j},p'\right) - H\left(2 - y\left(s6_{\text{in},j},p\right) - y\left(s6_{\text{in},j},p'-1\right)\right), \quad \forall p,p' \in P, p' > 1, p' \leq p, j = 2,3$$
$$t_u\left(s6_{\text{in},j},p\right) \geq t_p\left(s9_{\text{out},j},p'\right) - H\left(2 - y\left(s6_{\text{in},j},p\right) - y\left(s8_{\text{in},j},p'-1\right)\right), \quad \forall p,p' \in P, p' > 1, p' \leq p, j = 2,3$$
$$t_u\left(s8_{\text{in},j},p\right) \geq t_p\left(s6_{\text{out},j},p'\right) - H\left(2 - y\left(s8_{\text{in},j},p\right) - y\left(s2_{\text{in},j},p'-1\right)\right), \quad \forall p,p' \in P, p' > 1, p' \leq p, j = 2,3$$
$$t_u\left(s8_{\text{in},j},p\right) \geq t_p\left(s8_{\text{out},j},p'\right) - H\left(2 - y\left(s8_{\text{in},j},p\right) - y\left(s6_{\text{in},j},p'-1\right)\right), \quad \forall p,p' \in P, p' > 1, p' \leq p, j = 2,3$$
$$t_u\left(s8_{\text{in},j},p\right) \geq t_p\left(s9_{\text{out},j},p'\right) - H\left(2 - y\left(s8_{\text{in},j},p\right) - y\left(s8_{\text{in},j},p'-1\right)\right), \quad \forall p,p' \in P, p' > 1, p' \leq p, j = 2,3$$

This set of constraints ensures that each reaction commences after the completion of the other reactions, since they share the same units. The following constraints are also required to ensure that raw materials and products of each reaction are used and produced at the same point in time, respectively.

Reaction 1

$$t_u(s2_{\text{in},j},p) = t_u(s3_{\text{in},j},p), \forall p \in P, j = 2,3$$

Reaction 2

$t_u(s6_{\text{in},j}, p) = t_u(s5_{\text{in},j}, p), \forall p \in P, j = 2, 3$

$t_p(s7_{\text{out},j}, p) = t_p(s8_{\text{out},j}, p), \forall p \in P, j = 2, 3$

Reaction 3

$t_u(s8_{\text{in},j}, p) = t_u(s4_{\text{in},j}, p), \forall p \in P, j = 2, 3$

Assignment Constraint

As mentioned earlier, the fact that reactors 1 and 2 involve more than one task render the assignment constraint (constraint (2.19)) necessary so as to ensure that at any given time point only one task can commence. This is shown below.

$$y(s2_{\text{in},j}, p) + y(s6_{\text{in},j}, p) + y(s8_{\text{in},j}, p) \leq 1, \forall p \in P, j = 2, 3$$

The objective function for this formulation is the maximisation of revenue for products 1 and 2.

Computational Results

Table 2.5 gives the computational results for the second literature example.

Table 2.5 Results for the second literature example

	This approach	Ierapetritou and Floudas	Zhang	Schilling and Pantelides
NTP	5	6	7	6
NC	593	465	741	587
NV	400	310	497	386
NB	40	48	147	130
MILP solution	1513.35	1503.18	1497.69	1488.05
Relaxed objective	1735.53	1984.17	2258.71	2783.14
CPU time (s)	1.148	2.91	1027.5	N/A

NTP = number of time points; NC = number of constraints; NV = total number of variables; NB = number of binary variables

The results appearing in the second and third columns were obtained through the same processor and solver (600 MHz Pentium III processor; GAMS 2.5/OSL), albeit using SSN and STN based approaches, respectively. The other results were taken directly from literature (Ierapetritou and Floudas, 1998) and were obtained using different solvers from the one used in the second and third columns. Using 5 time points over 8-h time horizon, and modelling duration constraints as function

Table 2.6 Values for the binary variable at different time points for the second example

$y(s, p)$	p_1	p_2	p_3	p_4	p_5
$s1$	1	0	0	0	0
$s2_{in,2}$	1	0	0	0	0
$s2_{in,3}$	1	0	0	0	0
$s6_{in,2}$	0	1	0	1	0
$s6_{in,3}$	0	1	0	1	0
$s8_{in,2}$	0	0	1	0	0
$s8_{in,3}$	0	0	1	0	0
$s9$	0	0	0	1	0

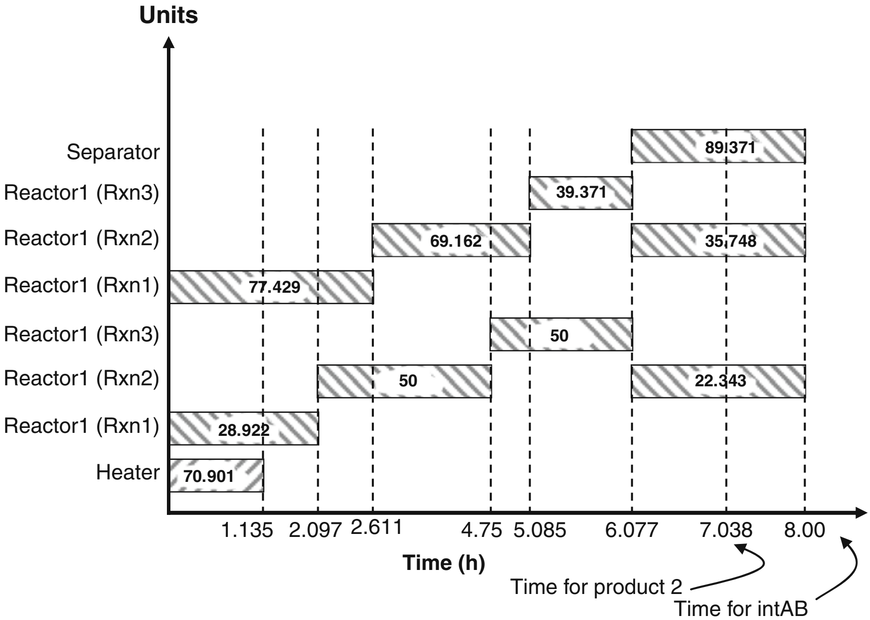


Fig. 2.8 Gantt chart for the second literature example

of batch size gave the objective value of 1513.35. Using more time points did not increase the objective value, but only increased the relaxed objective value from 1735.53 to 2054.68. It is worth noting that the proposed formulation requires only 40 binary variables compared to 48, 147 and 130, required in the formulations by Ierapetritou and Floudas (1998), Zhang (1995) and Schilling and Pantelides (1996), respectively. It should also be noted that, in this problem, the proposed formulation leads to a much smaller integrality gap, i.e. 12.8%, compared to the formulations by Ierapetritou and Floudas (24.24%), Zhang (33.7%) and Schilling and Pantelides (46.53%).

Table 2.6 shows the values of the binary variables at different time points. It is worthy of note that all the binary variables are zero at the end of the time horizon, i.e. p_5, implying that no state can be used at this point. The Gantt chart corresponding to these values of binary variables is shown in Fig. 2.8.

Note that in Fig. 2.8, product 2 is produced almost 1 h prior to the completion of separation in accordance with problem specification in Table 2.4.

2.5 Application of Aggregation Models in Reducing the Binary Dimension

The aggregation models are introduced in this chapter in order to demonstrate that, in operations where all the units involved in a particular stage have the same performance, the binary dimension can be dramatically reduced, thereby alleviating computational intensity. In using aggregation models, all the processors involved in a single stage are treated as a single unit. This implies that the use of aggregation models is concomitant with in-phase operation. It is, therefore, apparent that in a situation where optimality requires out-of-phase operation of units in a given stage, the use of an aggregation model will culminate in suboptimal results. However, experience with aggregation models has shown that, although they are a simplification, they give very comparable results to general models presented earlier, albeit with drastically reduced computational effort. This feature renders aggregation models potential candidates in an environment where quick solutions are sought, especially in terms of the objective value. The reduction in the number of binary variables is, in essence, consequent to the coupling of units in a stage, which eventually reduces the number of time points required, since all units in a given stage commence operations at the same time. To demonstrate the effectiveness of this concept, the second literature example is revisited.

2.5.1 Second Literature Example Revisited

Table 2.4 shows data for this example (Ierapetritou and Floudas, 1998). Since aggregation entails combination of all the processors in a particular stage, the main distinction from the general formulation is in the capacity constraints and the material balances. These are presented below.

Capacity Constraints

Reaction 1

States $s2$and $s3$producing $s6$

$$m_u(s2,p) + m_u(s3,p) \leq (50+80)y(s2,p), \forall p \in P$$

The coefficient of the binary variable is the sum of the capacities for the two reactors. State $s2$ has been chosen as the effective state.

Reaction 2

States $s5$ *and* $s6$ *producing* $s7$

$$m_u(s6, p) + m_u(s5, p) \leq (50 + 80)y(s6, p), \forall p \in P$$

In this reaction state $s6$ has been chosen as the effective state.

Reaction 3

States $s4$ *and* $s8$ *producing* $s9$

$$m_u(s8, p) + m_u(s4, p) \leq (50 + 80)y(s4, p), \forall p \in P$$

In this reaction state $s4$ has been chosen as the effective state.

Material Balances

Reaction 1

$$m_u(s2, p) + m_u(s3, p) = m_p(s6, p), \forall p \in P$$

Reaction 2

$$m_u(s6, p) + m_u(s5, p) = \frac{10}{4} m_p(s7, p), \forall p \in P$$

The coefficient of $m_p(s7, p)$ is taken from the mass fractions of product 1 and intAB given in the SSN.

Reaction 3

$$m_u(s8, p) + m_u(s4, p) = m_p(s9, p), \forall p \in P$$

It should be noted that, unlike in the general formulation, in all these constraints no distinction is made as to which unit the state goes in, since all units in a given stage are treated as a single unit.

Computational Results

Table 2.7 gives the computational results from the aggregation model.

The results shown in Table 2.7 display minor discrepancies between the objective values obtained using the aggregation model (column 2) and those obtained from the general model (column 3). The smaller number of both continuous and binary variables is a result of fewer operating units due to coupling of units in a given stage as aforementioned. As a consequence of its relatively reduced size, the aggregation model requires merely 0.438 CPU seconds to get the solution, whereas the general model requires 1.148 CPU seconds. Comparison of these results shows that even though the aggregation model is much smaller in size, it gives an objective value of

Table 2.7 Results for the second literature example using aggregation models

	This approach (AM)	This approach (GM)
NTP	5	5
NC	343	593
NV	236	400
NB	25	40
MILP solution	1498.257	1513.35
Relaxed objective	1730.652	1735.53
CPU time (s)	0.438	1.148

AM = aggregation model; GM = general model; NTP = number of time points; NC = number of constraints; NV = total number of variables; NB = number of binary variables

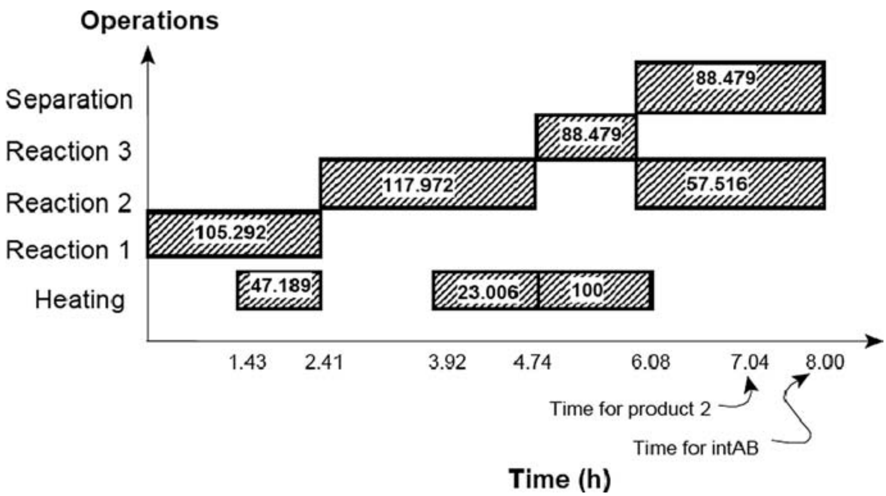


Fig. 2.9 Gantt chart for the second literature example using aggregation model

1498.257 which is over 99% close to 1513.35, as obtained from the general model. The Gantt chart corresponding to this formulation is shown in Fig. 2.9.

It is evident that the Gantt chart derived from an aggregation model does not show how much should allocated to each processor in stages with more than one processor, e.g. in reactions 1, 2 and 3. For an example, reaction 1 requires 105.292 capacity units between 0 and 2.41 h. To overcome this limitation of aggregation models, it should be realised that the performance of a reactor is not dependent on the capacity of the material, as long as it is within the design capacity limits. Therefore, this amount can be split into any feasible proportions to reactors 1 and 2, implying that both reactors should be conducting reaction 1 during this time interval. The same reasoning also applies to reactions 2 and 3. For example, if the capacity requirements for reactions 1, 2 and 3 are split into a 38:62 ratio between reactors 1 and 2, respectively, a schedule that is very similar to that given in Fig. 2.8 is obtained. The ratio that is mentioned in the foregoing statement is, in essence, the ratio of the reactor capacities given in the problem description (Table 2.4). Note that in Fig. 2.9, product 2 is produced almost 1 h prior to the completion of separation in accordance with problem specification in Table 2.4.

2.6 Summary and Conclusions

The main challenge in short-term scheduling emanates from time domain representation, which eventually influences the number of binary variables and accuracy of the model. Contrary to continuous-time formulations, discrete-time formulations tend to be inaccurate and result in an explosive binary dimension. This justifies recent efforts in developing continuous-time models that are amenable to industrial size problems.

In this chapter, state sequence network (SSN) representation has been presented. Based on this representation, a continuous-time formulation for scheduling of multipurpose batch processes is developed. This representation involves states only, which are characteristic of the units and tasks present in the process. Due to the elimination of tasks and units which are encountered in formulations based on the state task network (STN), the SSN based formulation leads to a much smaller number of binary variables and fewer constraints. This eventually leads to much shorter CPU times as substantiated by both the examples presented in this chapter. This advantage becomes more apparent as the problem size increases. In the second literature example, which involved a multipurpose plant producing two products, this formulation required 40 binary variables and gave a performance index of 1513.35, whilst other continuous-time formulations required between 48 (Ierapetritou and Floudas, 1998) and 147 binary variables (Zhang, 1995).

Lastly, this chapter presents the concept of aggregation as a means of reducing the binary dimension in large-scale problems. In the examples cited, the objective values predicted by the aggregation model were very close to those predicted by the general formulation. However, the aggregation model requires a much smaller number of binary variables which is concomitant with significantly reduced computational effort.

2.7 Exercise

Task: Determine the schedule that corresponds to maximum throughput over a 12 h time horizon

Problem description: Figure 2.10 is the flowsheet for the industrial case study used to illustrate the application of the SSN based approach. Figure 2.11 is the corresponding SSN.

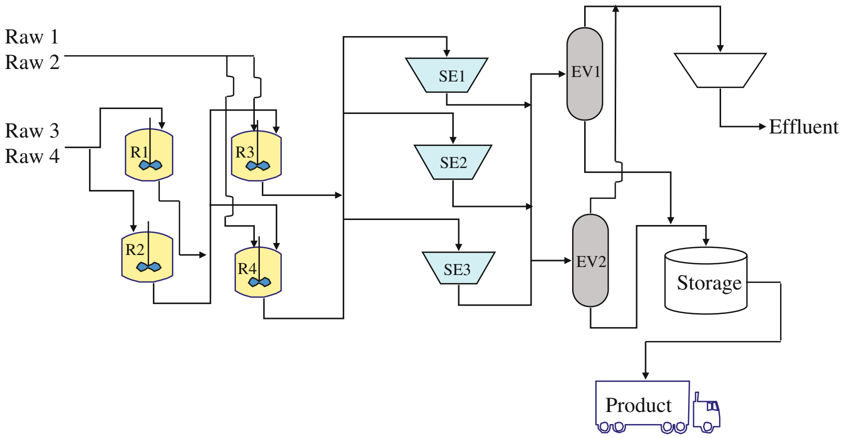


Fig. 2.10 Flowsheet for the industrial case study

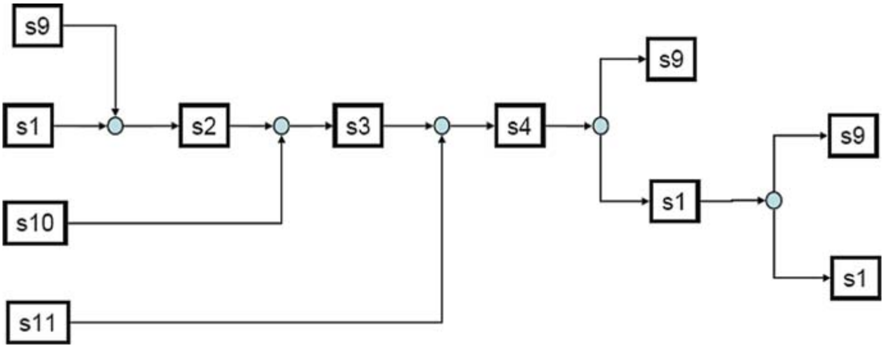


Fig. 2.11 SSN for the industrial case study

The process involved consists of 5 consecutive steps. The first step involves a reaction which forms an arsenate salt. This reaction requires two raw materials, raw 3 and raw 4, and can be conducted in either reactor R1 or R2. The arsenate salt from the first step is then transferred to either reactor R3 or R4 wherein two reactions take place. The first of these reactions is aimed at converting the arsenate salt to a disodium salt using raw material 1 (raw 1). The disodium salt is then reacted further to form the monosodium salt using raw material 2 (raw 2). The monosodium salt solution is then transferred to the settling step in order to remove the solid byproduct. Settling can be conducted in any of the three settlers, i.e. SE1, SE2 or SE3. The solid byproduct is dispensed with as waste and the remaining monosodium salt solution is transferred to the final step. This step consists of two evaporators, EV1 and EV2, which remove the excess amount of water from the monosodium solution. Evaporated water is removed as effluent and the monosodium salt (product) is taken to storage. States $s1$ and $s9$ in the SSN represent raw 3 and raw 4, respectively. States

$s10$ and $s2$ represent raw 1 and the arsenate salt, whilst states $s11$ and $s3$ represent raw 2 and the disodium salt, respectively. State $s4$ is the monosodium solution that is transferred to the settlers to form state $s8$ (solid byproduct) and state $s5$ (remaining monosodium solution). State $s5$ is separated into $s7$ (water) and $s6$ (product). Table 2.8 shows the data for the case study.

The stoichiometric data is included in order to perform material balances in each unit operation. The second column of the stoichiometric data shows the amount of raw material required (tons) per unit mass (tons) of the overall output, i.e. $s6 + s7 + s8$. The third column shows the ratio of each byproduct ($s7$ and $s8$) to product ($s6$) in ton/ton product. The objective function is the maximisation of product ($s6$) output. A 20% variation in processing times was assumed.

Table 2.8 Scheduling data for the industrial case study

Unit	*j*	Capacity	Suitability	Mean processing time (h)
R1	1	10	Reaction 1	2
R2	2	10	Reaction 1	2
R3	3	10	Reaction 2, reaction 3	3, 1
R4	4	10	Reaction 2, reaction 3	3, 1
SE1	5	10	Settling	1
SE2	6	10	Settling	1
SE3	7	10	Settling	1
EV1	8	10	Evaporation	3
EV2	9	10	Evaporation	3
State		Storage capacity		Initial amount
$s1$		Unlimited		Unlimited
$s2$		100		0
$s3$		100		0
$s4$		100		0
$s5$		100		0
$s6$		100		0
$s7$		100		0
$s8$		100		0
$s9$		Unlimited		Unlimited
$s10$		Unlimited		Unlimited
$s11$		Unlimited		Unlimited
		Stoichiometric data		
State		Ton/ton output	Ton/ton product	
$s1$		0.20		
$s9$		0.25		
$s10$		0.35		
$s11$		0.20		
$s7$			0.7	
$s8$			1	

References

Ierapetritou, M.G., Floudas, C.A., 1998. Effective continuous-time formulation for short-term scheduling. 1. Multipurpose batch processes. Ind. Eng. Chem. Res., 37: 4341–4359.

Schilling, G., Pantelides, C.C., 1996. A simple continuous-time process scheduling formulation and a novel solution algorithm. Comput. Chem. Eng., 20(Suppl.): S1221–S1226.

Zhang, X., 1995. Algorithms for optimal scheduling using nonlinear models. PhD thesis, University of London, London.

Chapter 3
Process Intermediate Storage Operational Philosophy: The New Operational Philosophy

Overview In Chapter 2 a mathematical formulation for scheduling that is based on the SSN recipe representation was presented. In this chapter this mathematical formulation is adapted to suit a novel operational philosophy, the Process Intermediate Storage (PIS) operational philosophy. Whilst exploiting this operational philosophy, the adapted formulation is used to synthesize, schedule and design multipurpose batch plants. Two cases are studied to determine the effectiveness of this operational philosophy. In the first case, which excludes any dedicated storage, the use of this operational philosophy results in 50% improvement in throughout. The second case is used to determine the minimum amount of intermediate storage while maintaining the throughput achieved with infinite intermediate storage. This results in 20% reduction in the amount of dedicated intermediate storage. In both cases the structure of the models is MILP. Furthermore, the MINLP design model is developed to exploit the attributes of the PIS operational philosophy.

3.1 The PIS Operational Philosophy

The models developed to take the PIS operational philosophy into account are detailed in this chapter. The models are based on the SSN and continuous time model developed by Majozi and Zhu (2001), as such their model is presented in full. Following this the additional constraints required to take the PIS operational philosophy into account are presented, after which, the necessary changes to constraints developed by Majozi and Zhu (2001) are presented. In order to test the scheduling implications of the developed model, two solution algorithms are developed and applied to an illustrative example. The final subsection of the chapter details the use of the PIS operational philosophy as the basis of operation to design batch facilities. This model is then applied to an illustrative example. All models were solved on an Intel Core 2 CPU, T7200 2 GHz processor with 1 GB of RAM, unless specifically stated.

The PIS operational philosophy is novel and thus requires further explanation. When a batch operation is scheduled a Gantt chart is usually generated, such as

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_3,

in Fig. 3.1. From this figure it is simple to identify the latent storage potential of the units. For example, units 1, 3, 4 and 6 are idle and empty for most of the time horizon of interest. This provides the opportunity of using these units as storage, instead of, or in conjunction with dedicated intermediate storage. This leads to a number of benefits, such as increased capital utilization of the equipment, possible reduction in the size required for the plant and a reduced capital cost associated with the construction of new batch facilities. In order to illustrate the idea even further, the following example was developed.

Figure 3.2 shows the flowsheet for the illustrative example. The data for the example is given in Table 3.1, where i_1, i_2, and i_3 are consecutive processing units, while $d_{2,3}$ is a dedicated intermediate storage vessel between processing units i_2 and i_3. The time horizon of interest in this example is 9 h.

When the PIS operational philosophy is not used, as in Fig. 3.3, 50 t of dedicated intermediate storage unit, $d_{2,3}$, was required. The reason for this is that the

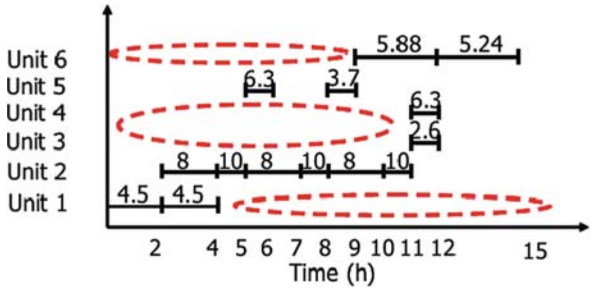


Fig. 3.1 General schedule

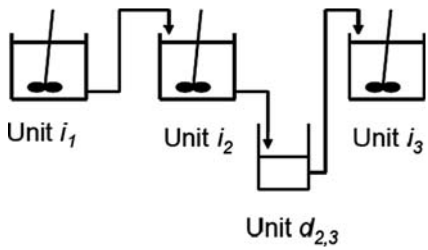


Fig. 3.2 Flowsheet for the simple example

Table 3.1 Data for simple illustrative example

Unit	Capacity (t)	Processing time
i_1	100	2
i_2	100	3
i_3	50	2
$d_{2,3}$	100	–

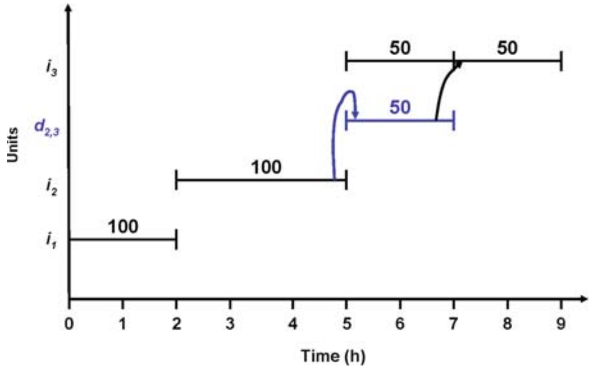


Fig. 3.3 Schedule of the simple example without using PIS

capacity of the final unit is only 50 t. Consequently, the 100 t batch produced from unit i_2 must be split in half. Half of the batch is stored in dedicated intermediate storage while the remaining batch is processed, after which, the stored mass is then processed thus achieving the optimal throughput of 100 t. However, when compared to the schedule in Fig. 3.4, the 100 t storage vessel is not needed. The reason for this is that 50 t of the intermediate product produced from unit i_2 is moved to i_1 for storage, while the remaining 50 t is processed in unit i_3. This increases the capital utilization of unit i_1, while reducing the size required for the plant which achieves the same throughput. This also avails unit i_2 for further processing. Furthermore, if this possibility had been identified at the design phase, the cost of the 100 t storage vessel could have been saved.

In order to illustrate the uses of this novel operational philosophy, i.e. PIS operational philosophy, this chapter has been divided into two parts. Firstly, the applicability of the operational philosophy will be proven and used to determine the minimum amount of intermediate storage required while maintaining the throughput achievable with infinite intermediate storage. Secondly, the PIS operational philosophy will be used to design storageless batch plants.

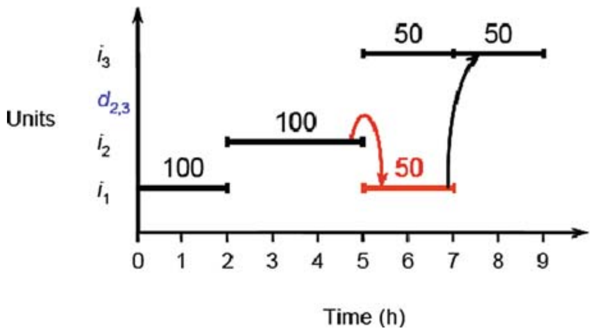


Fig. 3.4 Schedule of the simple example using PIS

3.2 Problem Statement

The problem pertaining to PIS philosophy can be formally stated as follows,
Given:

(i) the production recipe for each product, including processing times in each unit operation,
(ii) the available units and their capacities,
(iii) the maximum storage capacity for each material, and
(iv) the time horizon of interest.

determine,

(i) the maximum throughput with zero intermediate storage with and without using the PIS operational philosophy,
(ii) the minimum amount of intermediate storage, while maintaining the optimal throughput.

3.3 Mathematical Formulation

A mathematical model for the PIS philosophy is the adapted version of the mathematical formulation of Majozi and Zhu (2001) and composed of the following sets, variables and parameters.

Sets

$$P = \{p\,|p = \text{time point}\}$$
$$J = \{j\,|j = \text{unit}\}$$
$$S = \{s\,|s = \text{is any state}\}$$
$$S_{\text{in}} = \{S_{\text{in}}\,|S_{\text{in}} = \text{is any input state}\}$$
$$S_{\text{out}} = \{S_{\text{out}}\,|S_{\text{out}} = \text{is any output state}\}$$
$$S_{\text{in}}^{*} = \left\{S_{\text{in}}^{*}\,\middle|S_{\text{in}}^{*} = \text{effective state into unit}\right\} \subseteq S_{\text{in}}$$

Binary Variables

$y_{\text{in}}(S,j,p)$	decision variable describing the processing of states in unit j at time point p
$y^{\text{lt}}(s,j,p)$	decision variable describing the latent storage of state s in unit j at time point p
$e(j)$	decision variable based on whether unit j exists or not

Continuous Variables

$m_{\text{in}}(s,j,p)$	amount of state s consumed for processing in unit j at time point p
$m_{\text{in}}^{s}(s,j,p)$	amount of state s fed into storage from unit j at time point p
$m_{\text{in}}^{\text{lt}}\left(s,j,j',p\right)$	amount of state s fed into latent store j' from unit j at time point p
$m_{\text{out}}(s,j,p)$	amount of state s produced from unit j at time point p
$m_{\text{out}}^{s}(s,j,p)$	amount of state s fed from storage to unit j at time point p
$m_{\text{out}}^{\text{lt}}\left(s,j',j,p\right)$	amount of state s from latent store in j' fed to unit j at time point p
$c(j)$	capacity of unit j
$t_{\text{out}}(s,j,p)$	time at which state s is produced from unit j at time point p
$t_{\text{in}}(s,j,p)$	time at which state s is used in unit j at time point p
$t_{\text{in}}^{\text{lt}}(s,j,p)$	beginning of latent storage for state s in unit j at time point p
$t_{\text{out}}^{\text{lt}}(s,j,p)$	end of latent storage for state s in unit j at time point p
$w(s,j,p)$	storage time for state s in unit j during latent store at time point p
$q(s, p)$	amount of state s stored at time point p
$d(s, p)$	amount of state s delivered to customers at time point p
Z	objective function to be evaluated

Parameters

V_j^L	Minimum capacity of unit j
V_j^U	Maximum capacity of unit j
$Q_s^0(S)$	Amount of state initially stored
$T(s)$	Processing time of state s
$W^U(s,j)$	Upper limit of the duration of the latent storage of states in unit j
$W^L(s,j)$	Lower limit of the duration of the latent storage of states in unit j
H	Time horizon of interest

A, B	Unit specific capital cost terms
α	Power function for capital cost objective function
$Q_s^{\max}(s)$	maximum amount of state s stored within the time horizon of interest

Based on these variables and parameters, the following constraints apply. The reader's attention is drawn to the fact that the binary variable appears as a 3 index variable instead of a 2 index variable that was introduced in Chapter 2. This choice was deliberately made in order to simplify the overall formulation.

Capacity Constraints

$$V_j^L y\left(s_{\text{in}}^*, j, p\right) \leq \sum_{s \in S_{\text{in}}} m_{\text{in}}(s, j, p) \leq V_j^U y\left(s_{\text{in}}^*, j, p\right) \quad \forall s \in s_{\text{in},j}, j \in J, p \in P \quad (3.1)$$

Constraint (3.1) states that the mass entering a unit for processing must be between the minimum and maximum capacities of the unit. Furthermore, it ensures that if mass enters a unit, that unit becomes active.

Material Balances

$$\sum_{s \in S_{\text{in},j}} m_{\text{in}}(s, j, p-1) = \sum_{s \in S_{\text{out},j}} m_{\text{out}}(s, j, p) \quad \forall j \in J, p \in P, p > 1 \quad (3.2)$$

$$q(s,p) = q(s,p-1) + \sum_{j \in J_s^{\text{out}}} m_{\text{out}}(s,j,p) - \sum_{j \in J_s^{\text{in}}} m_{\text{in}}(s,j,p) \quad (3.3)$$
$$\forall s \in S, S = \text{ i ntermediate}, j \in J, p \in P, p > 1$$

$$q(s,p) = q(s,p-1) + \sum_{j \in J_s^{\text{in}}} m_{\text{in}}(s,j,p) \quad (3.4)$$
$$\forall s \in S, S = \text{feed}, j \in J, p \in P, p > 1$$

$$q(s,p) = q(s,p-1) + \sum_{j \in J_s^{\text{out}}} m_{\text{out}}(s,j,p) - d(s,p) \quad (3.5)$$
$$\forall s \in S, S = \text{product}, j \in J, p \in P, p > 1$$

$$q(s,p_1) = Q_s^o(s) - \sum_{j \in J_s^{\text{in}}} m_{\text{in}}(s,j,p_1) \quad \forall s \in S, j \in J, p_1 \in P \quad (3.6)$$

The material balances are shown by constraints (3.2), (3.3), (3.4), (3.5) and (3.6). Constraint (3.2) is a mass balance over a processing unit. It simply states that the mass that enters unit j at time point $p-1$ must exit that unit at the next time point. Constraint (3.3), is a balance over a dedicated intermediate storage unit and only applies to intermediate products. This constraints states that the amount of state s that is stored in the dedicated intermediate storage unit is the difference between that which enters and exits for processing and the amount of state s that was already present at the previous time point. Constraint (3.4) applies where mass is only used, such as with feed states. The amount of state delivered to the customer is determined by constraint (3.5), where a state is only produced not used. Constraint (3.6) is similar to constraint (3.3), however, it applies at the beginning of the time horizon of interest. This constraints takes care of the possibility that there is state stored in a unit prior to the start of scheduling, such as feeds or in the case where rescheduling is required.

Duration Constraints

$$t_{\text{out}}(s_{\text{out}},j,p) = t_{\text{in}}(s^*_{\text{in}},j,p-1) + \tau(s)y(s^*_{\text{in}},j,p-1) \quad \forall j \in J, p \in P, p > 1, s_{\text{out}} \in S_{\text{out}}, s_{\text{in}} \in S_{\text{in}} \tag{3.7}$$

The model is based on a fixed duration of tasks as shown in constraint (3.7). This constraints states that the time at which the output state from unit j exits, is the time at which the input state entered the unit at the previous time point plus the duration of the task. The binary variable ensures that the constraints holds whenever the unit is used at the precise time, i.e. $p-1$.

Sequence Constraints

$$t_{\text{in}}(s^*_{\text{in}},j,p) \geq \sum_{s_{\text{in}}} \sum_{s_{\text{out}}} \sum_{j \in J} \sum_{p' \in P} \left(t_{\text{out}}(s_{\text{out}},j,p') - t_{\text{in}}(s_{\text{in}},j,p-1)\right) \quad \forall j \in J, p \in P, p > 1, s_{\text{out}} \in S_{\text{out}}, s_{\text{in}} \in S_{\text{in}} \tag{3.8}$$

$$t_{\text{in}}(s^*_{\text{in}},j,p) \geq t_{\text{out}}(s_{\text{out}},j,p) \quad \forall j \in J, p \in P, s_{\text{out}} \in S_{\text{out}}, s_{\text{in}} \in S_{\text{in}} \tag{3.9}$$

$$lt_{\text{in}}(s_{\text{in}},j,p) \geq t_{\text{out}}(s_{\text{out}},j',p) \quad \forall j,j' \in J, p \in P, s_{\text{out}} = s_{\text{in}}, s_{\text{out}} \in S_{\text{out}}, s_{\text{in}} \in S_{\text{in}} \tag{3.10}$$

Constraint (3.8) reduces the search space by ensuring that the time at which a state s can be processed in unit j at time point p is at least after the sum of the durations of all previous tasks that have taken place in the unit. Constraint (3.9) ensures that the processing of state s_{in} into unit j can only take place after the previous batch has been processed. Constraint (3.10) stipulates that state sin can only be processed in unit j after it has been produced from unit j', where units j and j' are consecutive stages in the recipe.

3.3.1 Feasibility Constraints

$$\sum_{s \in S^*_{\text{in},j}} y(s,j,p) \leq 1 \quad \forall j \in J, p \in P \tag{3.11}$$

This constraints ensures that only one task can take place in a unit at a particular time point.

Time Horizon Constraints

$$t_{\text{in}}(s,j,p) \leq H \qquad \forall j \in J, p \in P, s \in S_{\text{in},j} \tag{3.12}$$

$$t_{\text{out}}(s,j,p) \leq H \qquad \forall j \in J, p \in P, s \in S_{\text{out},j} \tag{3.13}$$

Constraints (3.12) and (3.13) ensure that all the tasks take place within the time horizon of interest.

Storage Constraints

$$q\,(s,p) \leq Q_s^U\,(s) \quad \forall s \in S, p \in P \tag{3.14}$$

This constraints ensures that the maximum capacity of the intermediate storage units is not exceeded.

Extension to PIS Philosophy

The model in the above form does not take into account the possibility of using latent storage, i.e. PIS operational philosophy. There are a number of additional constraints needed to fully capture this operational philosophy.

Capacity Constraints

$$V_j^L y^{\mathrm{lt}}(s,j,p) \le \sum_{j' \in J} m_{\mathrm{in}}^{\mathrm{lt}}\left(s,j',j,p\right) \le V_j^U y^{\mathrm{lt}}(s,j,p) \quad \forall s \in S, j \in J, p \in P \tag{3.15}$$

Constraint (3.15) states that the mass entering a process unit for latent storage must be between the minimum and maximum capacities of the unit. Furthermore, it ensures that mass can only enter the unit if the binary variable associated with latent storage is active for unit j at time point p.

Material Balances

$$\sum_{j' \in J} m_{\mathrm{in}}^{\mathrm{lt}}(s,j',j,p-1) = \sum_{j' \in J} m_{\mathrm{out}}^{\mathrm{lt}}(s,j,j',p) \quad \forall s \in S, j \in J, p \in P, p > 1 \tag{3.16}$$

$$m_{\mathrm{in}}(s,j,p) = \sum_{j' \in J} m_{\mathrm{out}}^{\mathrm{lt}}(s,j',j,p) + m_{\mathrm{out}}^{\mathrm{s}}(s,j,p) \quad \forall s \in S, j \in J, p \in P \tag{3.17}$$

$$m_{\mathrm{out}}(s,j,p) = \sum_{j' \in J} m_{\mathrm{in}}^{\mathrm{lt}}(s,j',j,p) + m_{\mathrm{in}}^{s}(s,j,p) \quad \forall s \in S, j \in J, p \in P \tag{3.18}$$

The mass balance over a process unit which is being used as latent storage is given by constraint (3.16). It should be noted that the input and output states remain the same. Constraints (3.17) and (3.18) are the inlet and outlet mass balances for mass which is to be used for, or produced from processing, respectively. Mass which enters a unit for processing comes from dedicated storage and/or latent storage as stated in constraint (3.17). Similarly, mass which exits a unit is either moved to dedicated storage or latent storage as stated in constraint (3.18).

Duration Constraints

$$t_{\mathrm{out}}^{\mathrm{lt}}(s,j,p) = t_{\mathrm{in}}^{\mathrm{lt}}(s,j,p-1) + w(s,j,p) \quad \forall s \in S, j \in J, p \in P \tag{3.19}$$

$$W^L(s,j)y^{\mathrm{lt}}(s,j,p) \le \omega(s,j,p) \le W^U(s,j)y^{\mathrm{lt}}(s,j,p) \quad \forall s \in S, j \in J, p \in P \tag{3.20}$$

The duration constraints for a latent storage, constraint (3.19), is similar to constraint (3.7) except that the residence time is a variable in the latter case. In the case of latent storage the actual duration is a variable that can vary between the lower and upper limits specified for state s in unit j, as shown in constraint (3.20).

Sequence Constraints

$$t_{\text{in}}^{\text{lt}}(s,j,p) \geq t_{\text{out}}^{\text{lt}}(s',j,p') - H\left(2 - y^{\text{lt}}(s',j,p'-1) - y^{\text{lt}}(s,j,p)\right) \quad \forall s,s' \in S, \forall j \in J, \forall p,p' \in P, p \geq p', p' > 1 \tag{3.21}$$

$$t_{\text{in}}^{\text{lt}}(s,j,p) \geq t_{\text{out}}(s',j,p') - H\left(2 - y^{\text{lt}}(s,j,p) - y(s'',j,p'-1)\right) \quad \forall s,s' \in S, \forall j \in J, \forall p,p' \in P, p \geq p', p' > 1, s'' \to s' \tag{3.22}$$

$$t_{\text{in}}(s,j,p) \geq t_{\text{out}}^{\text{lt}}(s',j,p') - H\left(2 - y(s,j,p) - y^{\text{lt}}(s',j,p'-1)\right) \quad \forall s,s' \in S, \forall j \in J, \forall p,p' \in P, p \geq p', p' > 1 \tag{3.23}$$

$$t_{\text{out}}^{\text{lt}}(s,j,p) \leq t_{\text{in}}(s,j',p) + H\left(2 - y^{\text{lt}}(s,j,p-1) - y(s,j,p)\right) \quad \forall s \in S_{\text{in},j'}, j,j' \in J, p \in P, p > 1 \tag{3.24}$$

$$lt_{\text{out}}^{\text{lt}}(s,j,p) \geq t_{\text{in}}(s,j',p) - H\left(2 - y^{\text{lt}}(s,j,p-1) - y(s,j,p)\right) \quad \forall s \in S_{\text{in},j'}, j,j' \in J, p \in P, p > 1 \tag{3.25}$$

$$t_{\text{in}}^{\text{lt}}(s,j,p) \leq t_{\text{out}}(s,j',p) + H\left(2 - y^{\text{lt}}(s,j,p) - y(s',j',p-1)\right) \quad \forall s \in S_{\text{out},j'}, s' \in S_{\text{in},j'}, j,j' \in J, p \in P, p > 1 \tag{3.26}$$

$$t_{\text{in}}^{\text{lt}}(s,j,p) \geq t_{\text{out}}(s,j',p) - H\left(2 - y^{\text{lt}}(s,j,p) - y(s',j',p-1)\right) \quad \forall s \in S_{\text{out},j'}, s' \in S_{\text{in},j'}, s \in S, j,j' \in J, p \in P, p > 1 \tag{3.27}$$

$$t_{\text{in}}(s,j',p) \geq t_{\text{out}}(s,j,p) - H\left(2 - y(s,j',p) - y(s',j',p-1)\right) \quad \forall s \in S, j,j' \in J, p \in P, p > 1 \tag{3.28}$$

$$t_{\text{in}}(s,j',p) \leq t_{\text{out}}(s,j,p) + H\left(2 - y(s,j',p) - y(s',j',p-1)\right) \quad \forall s \in S, j,j' \in J, p \in P, p > 1 \tag{3.29}$$

Constraints (3.21), (3.22) and (3.23) ensure that a state can only be processed or stored in unit j when the unit is available. It is assumed that after a batch has been stored in a process unit then it must follow the next processing step in its recipe. Constraints (3.24) and (3.25) ensure that the time at which a state leaves a unit after latent storage coincides with the time that the state enters a unit which is capable of processing that state. Constraints (3.26) and (3.27), are similar to constraints (3.24) and (3.25), however these apply to a state moving from processing to latent storage. If mass is moved from processing in unit j to processing in unit j', the time at which the mass is produced must coincide with the time at which it is used, as shown by constraints (3.28) and (3.29).

Time Horizon Constraints

$$t_{\text{in}}^{\text{lt}}(s,j,p) \leq H \qquad \forall j \in J, p \in P, s \in S \tag{3.30}$$

$$t_{\text{out}}^{\text{lt}}(s,j,p) \leq H \qquad \forall j \in J, p \in P, s \in S \tag{3.31}$$

These constraints ensure that all storage activities take place within the time horizon of interest.

Feasibility Constraints

$$\sum_{s \in S_{\text{in}}} y^{\text{lt}}(s,j,p) + \sum_{s' \in S_{\text{in}}} y(s',j,p) \leq 1 \qquad \forall j \in J, p \in P \tag{3.32}$$

$$\sum_{j \in J} y^{\text{lt}}(s,j,p) \leq 1 \qquad \forall s \in S, p \in P \tag{3.33}$$

To ensure that a unit is only used for either processing or storage at a particular time point, constraint (3.32) is required. Constraint (3.33), ensures that a batch can not be split. Constraint (3.11) is redundant in the presence of constraint (3.32).

Necessary Modifications to the Basic Scheduling Model

In order to ensure the completeness of the model that takes the PIS operational philosophy into account, the basic scheduling model developed by Majozi and Zhu (2001) has to be modified as follows.

$$q(s,p) = q(s,p-1) + \sum_{j \in J_s^{\text{out}}} m_{\text{in}}^{s}(s,j,p) - \sum_{j' \in J_s^{\text{in}}} m_{\text{out}}^{s}(s,j',p) \qquad \forall s \in S, j \in J, p \in P \tag{3.34}$$

$$q(s,p) = q(s,p-1) - \sum_{j' \in J_s^{\text{in}}} m_{\text{out}}^{s}(s,j',p) \qquad \forall s \in S, S = \text{feed}, j \in J, p \in P, p > 1 \tag{3.35}$$

$$q(s,p) = q(s,p-1) + \sum_{j \in J_s^{\text{out}}} m_{\text{in}}^{s}(s,j,p) - d(s,p) \qquad \forall s \in S, S = \text{product}, j \in J, p \in P, p > 1 \tag{3.36}$$

$$q(s,p_1) = Q_s^{o}(s) - \sum_{j' \in J_s^{\text{in}}} m_{\text{out}}^{s}(s,j,p_1) \qquad \forall s \in S, j \in J, p_1 \in P \tag{3.37}$$

The balance over a dedicated intermediate storage unit has to be modified because of the possibility of latent storage. Constraint (3.34), provides the link for the inlet and outlet mass balance between units, as shown in constraints (3.17) and (3.18). Constraints (3.35), (3.36) and (3.37), are similar to constraints (3.4), (3.5) and (3.6), however they apply to the case where the PIS operational philosophy is taken into account.

$$t_{\text{in}}\left(s_{\text{in}}, j, p\right) \geq \sum_{s} \sum_{j} \sum_{p' \leq P} \begin{pmatrix} t_{\text{out}}\left(s_{\text{out}}, j, p'\right) - t_{\text{in}}\left(s_{\text{in}}, j, p' - 1\right) \\ + t_{\text{out}}^{\text{lt}}\left(s', j, p'\right) - t_{\text{in}}^{\text{lt}}\left(s', p' - 1\right) \end{pmatrix} \tag{3.38}$$

$$\forall j \in J, p \in P, p > 1, p' > 1, s_{\text{out},j} \in S_{\text{out},j}, s_{\text{in},j} \in S_{\text{in},j}, s' \in S$$

Constraint (3.8) has to be modified to include the possibility of using a unit as latent storage, as shown by constraint (3.38).

Possible Performance Indices (Objective Functions)

The main goal of the project is the minimization of plant size via the exploitation of latent storage. In order to achieve this goal two cases are considered. The goal of the first case is to check the advantages gained in terms of throughput (constraint (3.39)) when there is zero intermediate storage (constraint (3.40)), while in the second case the goal is the minimization of intermediate storage (constraint (3.43)) while maintaining the optimal throughput (constraint (3.44)). In this case the optimal throughput is defined as that which is achieved when the model is solved with infinite intermediate storage. Both cases investigate the effect of using latent storage.

Case 1

$$Z = \max \sum_{s \in S} \sum_{p \in P} d(s, p) \tag{3.39}$$

while

$$q(s, p) = 0 \qquad \forall s \in S, p \in P \tag{3.40}$$

Case 2

Step 1:

$$Z = \max \sum_{s \in S} \sum_{p \in P} d(s, p) \tag{3.41}$$

while

$$q(s, p) > 0 \qquad \forall s \in S, p \in P \tag{3.42}$$

Step 2:

$$Z = \min \sum_{s \in S} \sum_{p \in P} d(s,p) \tag{3.43}$$

while

$$d(s,p) = \text{production goal} \qquad \forall s \in S, p \in P \tag{3.44}$$

3.4 Illustrative Examples

In this section, 2 illustrative examples are presented to demonstrate the applicability of this mathematical formulation.

3.4.1 First Illustrative Example

In order to illustrate these cases an example taken from the publications of Ierapetritou and Floudas (1998) and Majozi and Zhu (2001) will be used. The flow-sheet for the example is given in Fig. 3.5 with the SSN representation shown in Fig. 3.6. The data for the example is shown in Table 3.2, the time horizon of interest for this example has been altered from 12 h presented in Ierapetritou and Floudas (1998) and Majozi and Zhu (2001) to 24 h for illustrative purposes.

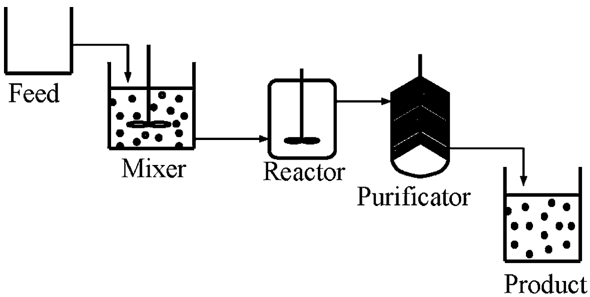


Fig. 3.5 Flowsheet for the literature example

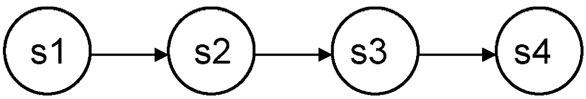


Fig. 3.6 SSN for the literature example

Table 3.2 Data for illustrative example

Unit	Capacity	Suitability	Processing time (h)
1	100	Mixing	4.5
2	75	Reaction	3
3	50	Purification	1.5

State	Storage capacity	Initial amount	Price
1	Unlimited	Unlimited	0.0
2	100	0.0	0.0
3	100	0.0	0.0
4	Unlimited	0.0	1.0

Case 1

Case 1 involves solving the model using the first case where the objective is to find the maximum throughput with zero intermediate storage without using the PIS operational philosophy and then resolving the model with the use of the PIS operational philosophy to compare the results. The optimal throughput achieved without using the PIS operational philosophy is 200. The schedule is shown in Fig. 3.7. The stair step nature of the schedule is expected due to the NIS operational philosophy. When the PIS operational philosophy is introduced, i.e. addition of constraints (3.15)–(3.38) to constraints (3.1)–(3.14), the optimal throughput increases from 200 to 300 units as shown in Fig. 3.8.

The way this improvement is achieved is clearly seen in Fig. 3.8. A portion of a batch is stored in a unit while the remaining batch is sent to processing. In this case half of the batch processed in the mixer is taken for storage in the purificator while the remaining mass is processed in the reactor. Once the reaction has proceeded to completion the mass that was stored in the purificator is moved to the reactor for processing. Further latent storage is required at 13.5 and 18 h, in this case the mass is stored in the reactor after processing and then moved to the purificator for

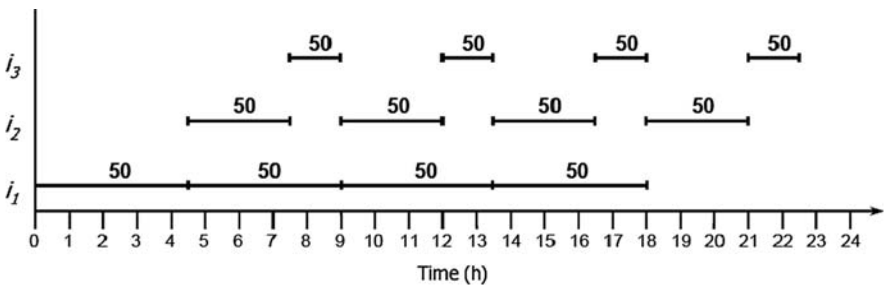


Fig. 3.7 Literature example without using the PIS operational philosophy

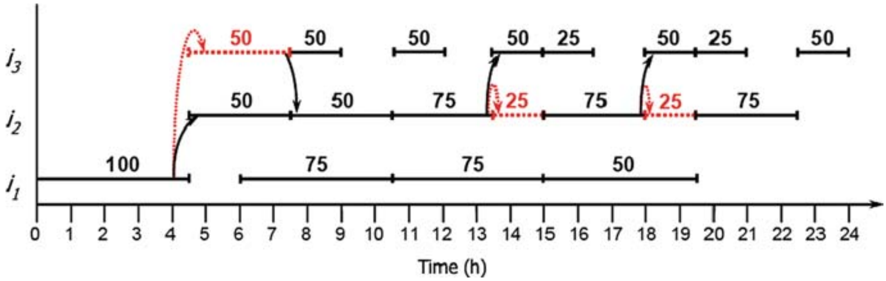


Fig. 3.8 Literature example using the PIS operational philosophy

Table 3.3 Results from the first case

	Without PIS philosophy	With PIS philosophy
Number of time point	7	10
Objective function value	200	300
Number of binary variables	18	87
Solution time (CPU s)	0.062	1.718
Number of variables	561	921
Number of constraints	1619	2592

processing. It should be noted that there is a cycle in this schedule. A cycle occurs when a unit fills and empties at the same time.

The model was solved using GAMS and the CPLEX solver version 9.1.2. The computational results for case 1 are shown in are shown in Table 3.3. From these results it is clear to see the potential benefits for using the PIS operational philosophy, with a 50% increase in throughput.

Case 2

The purpose of this case is to determine the minimum amount of intermediate storage available while maintaining the optimal throughput, where the optimal throughput is defined as that which is achieved when the model is solved with infinite intermediate storage. In this example the optimal throughput was 350 units. The schedule for this case is shown in Fig. 3.9. Without using the PIS operational philosophy the minimum amount of intermediate storage required was 250 units. The schedule for this case is shown in Fig. 3.10. In the case where the PIS operational philosophy was used a minimum of 200 units of dedicated intermediate storage was required. The schedule for this example is shown in Fig. 3.11.

The model was solved on an Intel Core 2 CPU, T7200 2 GHz processor with 1 GB of RAM, using GAMS and the CPLEX solver version 9.1.2. The computational results for the second case are shown in Table 3.4.

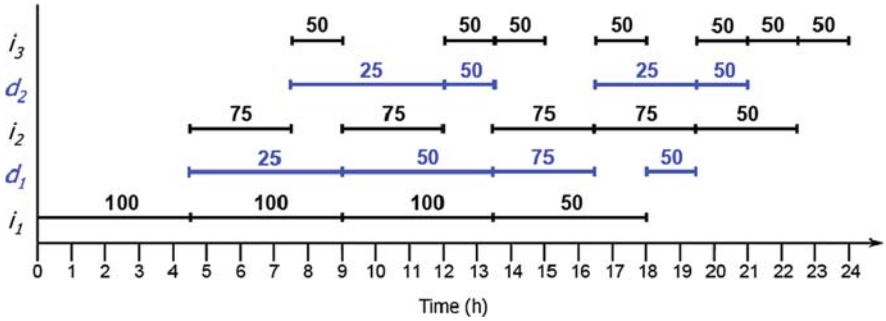


Fig. 3.9 Literature example with infinite intermediate storage

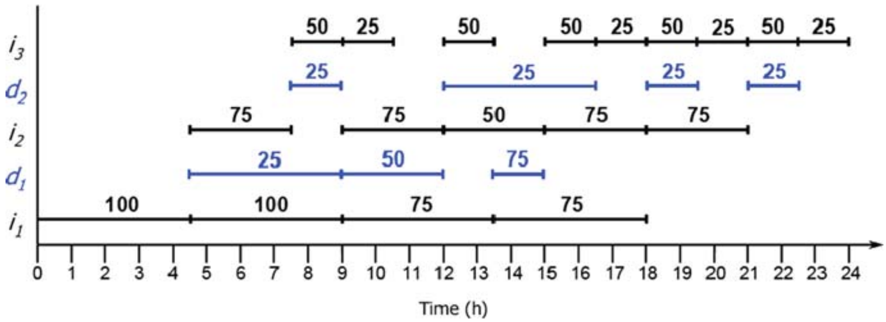


Fig. 3.10 Literature example without using the PIS operational philosophy

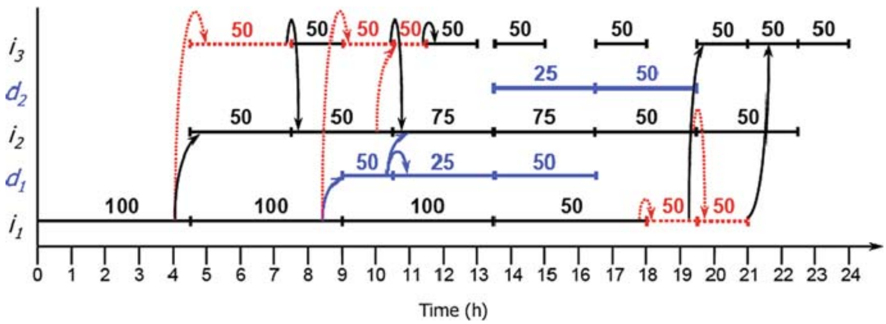


Fig. 3.11 Literature example using the PIS operational philosophy

Table 3.4 Results from the second case

	Infinite storage	Without PIS	With PIS
Number of time point	10	13	11
Objective function value	350[a]	250[b]	200[c]
Number of binary variables	27	36	96
Solution time (CPU s)	0.156	5.734	10.125
Number of variables	801	605	1014
Number of constraints	2432	2605	2893

[a]maximum throughput; [b]minimum storage without PIS; [c]minimum storage with PIS

Design Implications

This section furthers the model developed in the previous section to include the possibility of design.

Problem Statement

The problem considered in this subsection can be stated as follows,
Given:

(i) the production recipes, i.e. processing times for each task in a suitable unit as well as their sequence,
(ii) the availability and suitability of process vessels,
(iii) the potential number of process units in a stage, and range of capacity of potential process vessels,
(iv) production requirement, and
(v) the time horizon of interest.

determine:

the optimal number of units in a particular stage so as to minimise the capital cost.

Necessary Modifications to Case 1

Constraints (3.1), (3.2), (3.7), (3.9), (3.10), (3.12), (3.13), (3.15)–(3.31), (3.34), (3.35), (3.36), (3.37) and (3.38) still apply to the model, however, further modifications of the model are required to take into account the possibility of design.

Capacity Constraints

$$V_j^L e\,(j) \le c\,(j) \le V_j^U e\,(j) \qquad \forall j \in J, p \in P \tag{3.45}$$

$$m_{\text{in}}^{\text{lt}}(s,j,j',p) \le c(j') \qquad \forall s \in S, j, j' \in J, p \in P \tag{3.46}$$

$$m_{\text{in}}(s,j,p) \leq c(j) \qquad \forall s \in S, j \in J, p \in P \tag{3.47}$$

$$c\,(j) \leq c\left(j'\right) + V_j^U\left(2 - e\,(j) - e\left(j'\right)\right) \qquad \forall j, j' \in J \tag{3.48}$$

$$c\,(j) \geq c\left(j'\right) - V_j^U\left(2 - e\,(j) - e\left(j'\right)\right) \qquad \forall j, j' \in J \tag{3.49}$$

Constraint (3.45) ensures that the capacity of unit j is between the minimum and maximum permissible range, furthermore, it ensures that for a unit to have a capacity it must exist. Constraint (3.46) ensures that the mass entering unit j' for latent storage does not exceed the capacity of the unit. Constraint (3.47) is similar to constraint (3.46), however, it applies to unit j which is processing state s at time point p. It is further assumed that units in the same stage all have the same capacity, as shown by constraints (3.48) and (3.49), where units j and j' are units in the same stage.

Feasibility Constraint

$$\sum_{s \in S} y(s,j,p) + \sum_{s' \in S} y^{\text{lt}}(s',j,p) \leq e(j) \qquad \forall j \in J, p \in P \tag{3.50}$$

Constraint (3.50) is similar to constraint (3.32), however, it ensures that a unit can only be used for either processing or latent storage if that unit exists.

The objective function, which in this case is the minimization of capital cost, is shown in constraint (3.51). Due to the non-linearity of this constraint the model becomes an MINLP model.

$$\sum_{j \in J} \left(Ae(j) + B\left[c(j)\right]^{\alpha}\right) \tag{3.51}$$

3.4.2 Second Illustrative Example

This example is similar to the previous example in Section 3.4.1, however, there is another reactor in the reaction stage as shown in Fig. 3.12. The SSN remains the same and is shown here in Fig. 3.13. The data for this example is shown in Tables 3.5 and 3.6.

Results

The model was solved using GAMS DICOPT, with CLPEX as the MIP solver and CONOPT as the NLP solver. The computational results are shown in Table 3.7. The resulting plant requires only one reactor as shown in Fig. 3.14. The optimal capacities of the remaining units are 75 units for the mixer ($U1$), 75 units for the reactor

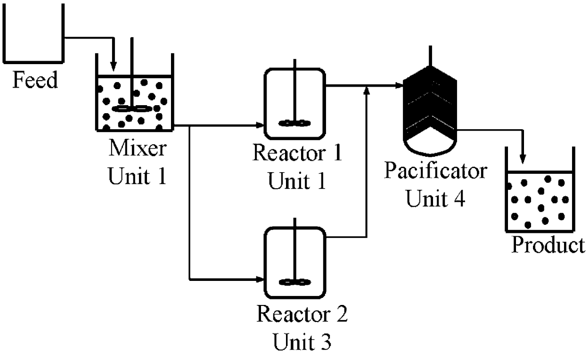


Fig. 3.12 Flowsheet for the literature example

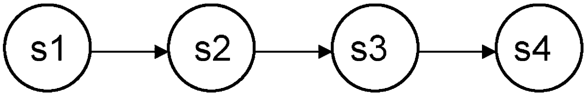


Fig. 3.13 SSN for the literature example

Table 3.5 Design data for illustrative example

Unit	Capacity range	Suitability	Processing time	Capital cost
1	25–100	Mixing	4.5	$V^{0.68}$
2, 3	25–75	Reaction	3	$V^{0.6}$
4	25–50	Purification	1.5	$V^{0.7}$

Table 3.6 Design data for illustrative example

State	Storage capacity	Initial amount	Price
1	Unlimited	Unlimited	0.0
2	0.0	0.0	0.0
3	0.0	0.0	0.0
4	Unlimited	0.0	1.0

Table 3.7 Computational results of the design literature example

	Results
Number of time points	11
Number of constraints	3864
Number of variables	1616
Number of binary variables	132
MINLP solution	44.82
CPU time (s)	35.858
Number of major iterations	3

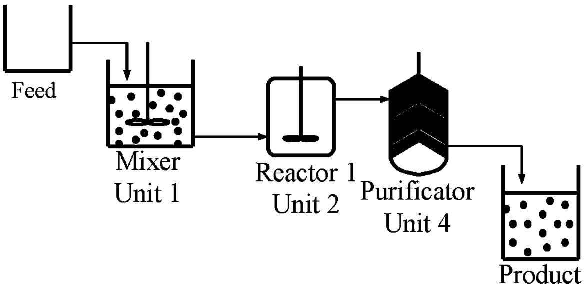


Fig. 3.14 Resulting design from the model

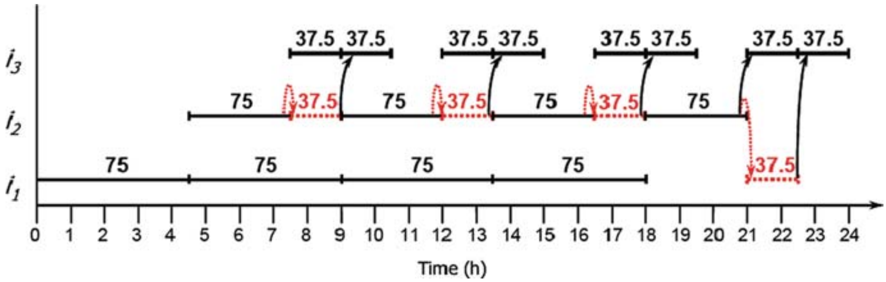


Fig. 3.15 Schedule for the optimal design

(*U*2) and 37.5 units for the purificator (*U*3). The resulting schedule for the optimal plant is shown in Fig. 3.15, where the numbers above the bars are the amount of each state processed and the dotted lines represent the storage of a state in a process unit.

Conclusions on the Results for the Illustrative Examples

MILP and MINLP models have been developed to take into account the PIS operational philosophy for assessing the efficacy of this philosophy and design, respectively. The MILP model is used to determine the effectiveness of the PIS operational philosophy by, firstly, solving the model with zero intermediate storage with and without the use of latent storage. In the illustrative example a 50% increase in the throughput was achieved.

Secondly, the minimum amount of intermediate storage is determined with and without the PIS operational philosophy. In both cases the production goal was set to that which was achieved when the model was solved with infinite intermediate storage. In the illustrative example a 20% reduction in the amount of intermediate storage is achieved. The design model is an MINLP model due to the non-linear capital cost objective function. This model is applied to an illustrative problem and results in the flowsheet as well as determining the capacities of the required units.

3.5 Industrial Application

The industrial case study presented in this section is taken from the petrochemicals industry. The project is in the design phase and as such the design model will be used to determine the design which leads to the minimum capital cost, while using the PIS operational philosophy. For secrecy reasons the example has been modified and the names of the raw materials and products have been changed to the generic form.

The flowsheet for the industrial case study is shown in Fig. 3.16. This case study is used to illustrate the application of the design model. The SSN for the case study is shown in Fig. 3.17.

The process has four stages, separated in Fig. 3.16 by dashed lines. The first stage involves the reaction between raw 1 and raw 2. This reaction can take place in either of the two reactors (R1_1 and R1_2) in stage 1. The intermediate produced in this reaction is then transferred to any of the four reactors in the second step (R2_1, R2_2, R2_3 and R2_4) where a further reactant, raw 3 is added as well as the solvent. In this stage the hot solvent is added to the reactor, so parts of the reaction mixture from the previous reaction are flashed off and sent to the scrubber. After 3 h of drying, raw 3 is added and the reaction proceeds. During the reaction, parts of the reaction mixture are vented and transported to storage tank, Stor_1, before distillation in unit D1. In the distillation stage the raw material, raw 1, is separated from the solvent. Following the separation both the raw 1 and the solvent are recycled

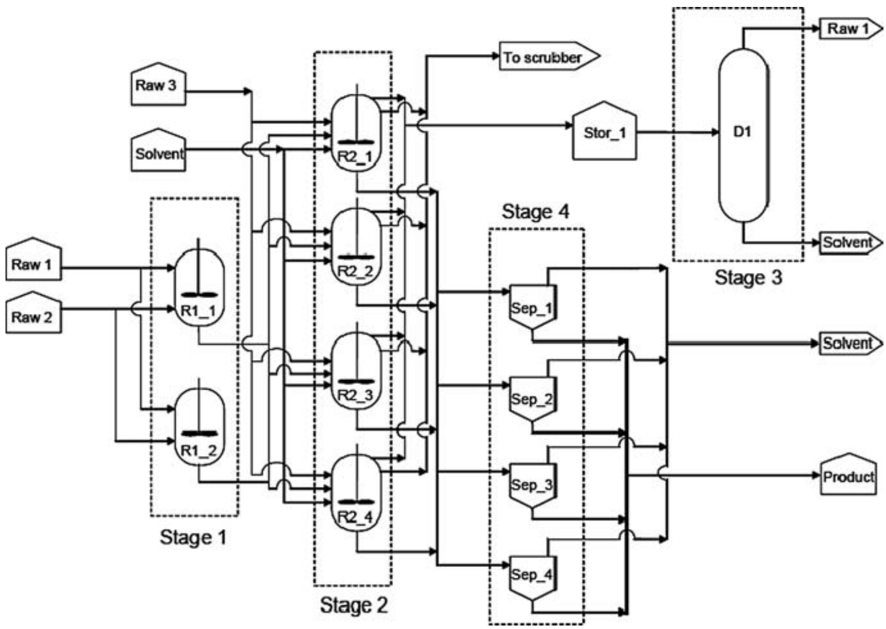


Fig. 3.16 Flowsheet for the industrial case study

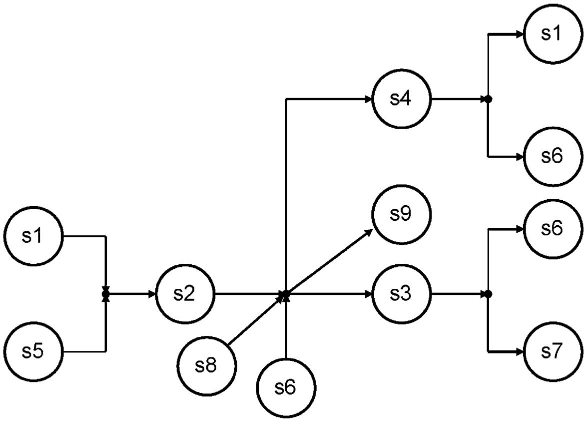


Fig. 3.17 SSN for the industrial case study

back to storage for reuse. The remaining reaction mixture is then transferred to any of the four settlers (Sep_1, Sep_2, Sep_3 and Sep_4) where the product is separated from the solvent. The solvent is then recycled back to the solvent storage tank to be reused. All of the units in stages 1, 2 and 4 can be used for latent storage, while, the distillation column, unit D1, in stage 4 cannot be used as storage for contamination reasons. Furthermore, only intermediate states can make use of latent storage. The time horizon of interest for the case study is 48 h for illustrative purposes. The production goal for the plant was 109.9 t of product.

The data for the case study is shown in Tables 3.8, 3.9, 3.10, and 3.11.

Table 3.8 Data for industrial case study

Unit	Max capacity (t)	Suitability	Duration (h)
R1_1, R1_2	25	Reaction 1	5
R2_1, R2_2, R2_3 and R2_4	83	Reaction 2	8
D1	100	Purification	1
Sep_1, Sep_2, Sep_3 and Sep_4	41	Separation	4

Table 3.9 Capital cost data for industrial case study

Unit	Capital cost
R1_1, R1_2	$89.1(c(j)/V_j^U)^{0.6}$
R2_1, R2_2, R2_3 and R2_4	$169.5(c(j)/V_j^U)^{0.6}$
D1	$41.4(c(j)/V_j^U)^{0.6}$
Sep_1, Sep_2, Sep_3 and Sep_4	$109(c(j)/V_j^U)^{0.6}$

Table 3.10 Storage and initial amount of state for the case study

State	Description	Storage capacity	Initial amount
S1	Raw 1	500	400
S2	Stage 2 feed	0	0.0
S3	Stage 4 feed	0	0.0
S4	Stage 3 feed	25	0.0
S5	Raw 2	400	400
S6	Solvent	1000	100
S7	Product	600	0.0
S8	Raw 3	Unlimited	Unlimited
S9	Vent to scrubber	Unlimited	0.0

Table 3.11 Feed and output ratios

State	Units	Ton/ton
S1/S5	R1_1, R1_2	0.9
S2/S8	R2_1, R2_2, R2_3 and R2_4	7
S2/S6	R2_1, R2_2, R2_3 and R2_4	0.5
S3/S9	R2_1, R2_2, R2_3 and R2_4	4
S3/S4	R2_1, R2_2, R2_3 and R2_4	15
S6/S7	Sep_1, Sep_2, Sep_3 and Sep_4	3.5
S1/S6	D1	0.02

3.5.1 Computational Results

The model was solved on an Intel Core 2 CPU, T7200 2 GHz processor with 1 GB of RAM. The computational results are shown in Table 3.12. The model was solved using GAMS DICOPT using CONOPT as the NLP solver and CPLEX as the MIP solver. The resultant flowsheet is shown in Fig. 3.18, as can be seen from this flowsheet the design calls for fewer units in stages 2 and 4. The schedule is shown in Fig. 3.19. As can be seen from the schedule, latent storage is utilised during the

Table 3.12 Results from the industrial case study

	Results
Number of time points	8
Objective function value	1727.9
Number of binary variables	228
Solution time (CPU s)	12082
Number of variables	4777
Number of constraints	8183
Number of major iterations	3

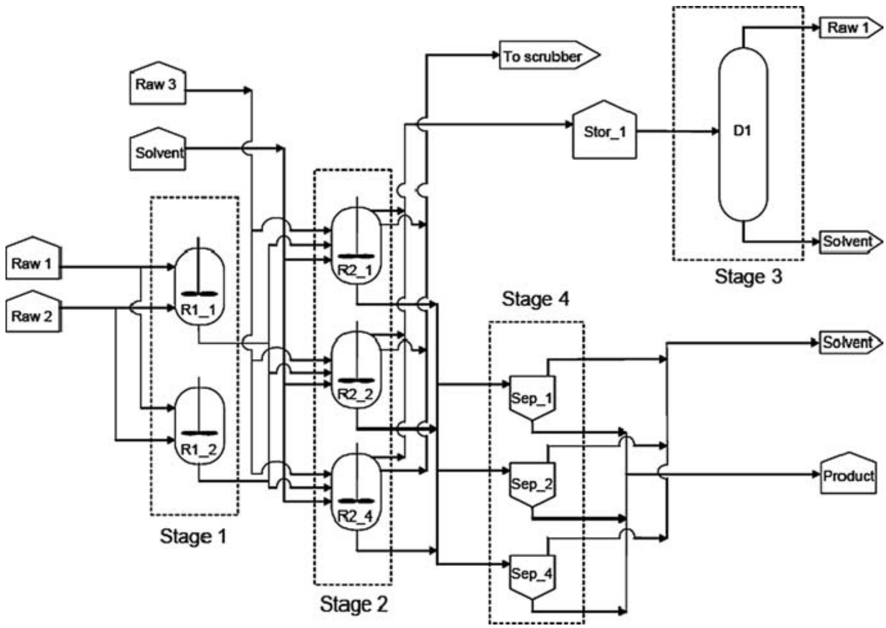


Fig. 3.18 Resultant flowsheet for the industrial case study

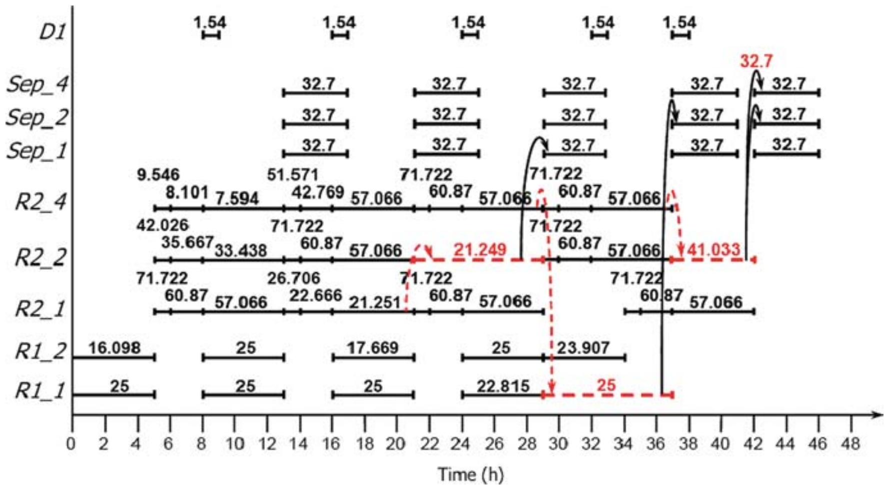


Fig. 3.19 Schedule for the optimal design

Table 3.13 Unit capacity results from the industrial case study

Unit	Capacity
R1_1, R1_2	25
R2_2, R2_3 and R2_4	75.138
Sep_1, Sep_2 and Sep_3	32.700
D1	1.54

time horizon of interest. Table 3.13 details the required unit capacities, which are lower than the original design.

3.5.2 Conclusions and Discussion on the Industrial Application

The model is successfully applied to the case study resulting in fewer units required to meet the demand. Furthermore, the units that are required have a lower capacity than the original design called for. The model also makes effective use of the latent storage available during the time horizon of interest. It is clear from Table 3.12 that the solution time for this case study is long. The reasons for this are that there is a large degree of complexity due to the possible use of latent storage. The use of latent storage also has a significant contribution on the overall number of binary variables, which can lead to increases in solution times. Due to the nonlinearity of the objective function global optimality is not assured.

3.6 Conclusions

The model developed was based on the framework and model developed by Majozi and Zhu (2001) for the following reasons. Firstly, the models that exploit the structure of the SSN result in fewer binary variables than those derived from other mathematical methods, because the SSN only takes states into account while tasks are implicitly incorporated. Secondly, this model is based on the non-uniform discretization of the time horizon, thus resulting in fewer binary variables. Thirdly, the model is a MILP, thus solutions are globally optimal.

Two distinctive models were developed in order to investigate the effectiveness of PIS operational philosophy. The first model is separated into two parts. The first part is used to determine the optimal throughput when there is zero intermediate storage available. Two situations were studied. Firstly the model was solved without the use of the PIS operational philosophy. Secondly, the model was solved with the PIS operational philosophy. In the simple example shown in this section a 50% increase in the throughput was achieved when the PIS operational philosophy was used. In both cases the models developed were a MILP, thus guaranteeing global optimality.

The second part of the first model was used to determine the minimum amount of intermediate storage required to achieve the same throughput achieved when there is infinite storage available. This part required a three step algorithm. Firstly, the optimal throughput was determined where there was infinite storage available. Secondly, the model was solved with the objective of minimizing the amount of intermediate storage without the use of the PIS operational philosophy, whilst keeping the optimal throughput from the first step fixed. The third step of the algorithm is similar to the second, however, the PIS operational philosophy is used. In the literature example an optimal throughput of 350 units was achieved in the first step. Using this as the fixed objective, a 20% reduction in the amount of required intermediate storage was achieved in the PIS operational philosophy compared to the case without the PIS operational philosophy. Once again the models derived were MILP models, thus guaranteeing global optimality.

The second mathematical formulation presented, is a design model based on the PIS operational philosophy. This formulation is an MINLP model due to the capital cost objective function. The model is applied to a literature example and an improved design is achieved when compared to the flowsheet. The design model is then applied to an industrial case study from the phenols production facility to determine its effectiveness. The data for the case study are subject to a secrecy agreement and as such the names and details of the case study are altered.

The model is successfully applied to the case study resulting in lower capacity and fewer units than the original design while achieving the required production goal. The model also makes effective use of the latent storage available during the time horizon of interest. However, the solution time for this case study is long, due to the large number of binary variables and a complex non-linear objective function which leads to a high degree of complexity. Furthermore, the possible use of latent

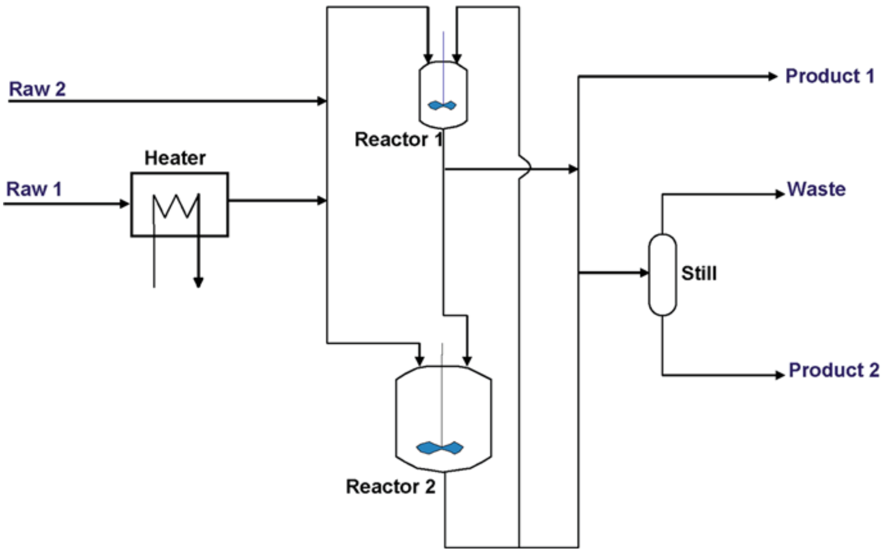


Fig. 3.20 Flowsheet for the second literature example

Table 3.14 Data for the second literature example; BATCH1

Unit	Capacity	Suitability	Mean processing time (τ)
Heater ($j = 1$)	100	Heating	1.0
Reactor 1 ($j = 2$)	50	Reaction 1, 2, 3	2.0, 2.0, 1.0
Reactor 2 ($j = 3$)	80	Reaction 1, 2, 3	2.0, 2.0, 1.0
Still ($j = 4$)	200	Separation	1 for product 2, 2 for IntAB
State	Storage capacity	Initial amount	Price
Feed A	Unlimited	Unlimited	0.0
Feed B	Unlimited	Unlimited	0.0
Feed C	Unlimited	Unlimited	0.0
Hot A	100	0.0	0.0
IntAB	200	0.0	0.0
IntBC	150	0.0	0.0
Impure E	200	0.0	0.0
Product 1	Unlimited	0.0	10.0
Product 2	Unlimited	0.0	10.0

storage results in a large number of possible combinations which also contribute to the overall complexity of the model.

3.7 Exercise

Task: Maximize revenue in the BATCH1 example presented in Chapter 2 through exploitation of PIS philosophy for both reactors over an 8 h time horizon.

Problem description: Figure 3.20 shows the flowsheet for BATCH1 example and Table 3.14 gives the scheduling data (reproduced from Chapter 2).

References

Ierapetritou, M., Floudas, C., 1998. Effective continuous-time formulation for short-term scheduling. I. Multipurpose batch processes. Ind. Eng. Chem. Res., 37(11): 3037–3051.

Majozi, T., Zhu, X., 2001. A novel continuous-time MILP formulation for multipurpose batch plants. I. Short-term scheduling. Ind. Eng. Chem. Res., 40(23): 5935–5949.

Chapter 4
Wastewater Minimisation in Multiproduct Batch Plants: Single Contaminants

Overview The previous 3 chapters gave some background on the fundamental aspects of batch processes with a particular focus on capturing the essence of time. No particular emphasis was placed on process integration aspects, like wastewater minimisation through recycle and reuse, as well as heat integration. Starting with this chapter, the rest of this textbook will address aspects of process integration. In this chapter, a mathematical formulation for freshwater and wastewater minimization in multipurpose batch plants with and without a central reusable water storage facility is presented. This formulation is based on uneven discretization of the time horizon as introduced in Chapter 1. The formulation that is presented in this chapter is targeted at multiproduct rather than multipurpose facilities. A fully generalized model that is applicable to both multiproduct and multipurpose batch plants is presented in Chapter 6 of this textbook.

Minimization of wastewater is achieved through the exploitation of recycle and reuse opportunities using a superstructure that entails all the possible recycle and reuse options. In terms of handling time, two broad cases are considered. In the first case the formulation is based on a predefined schedule, which implies that starting and finishing times are specified a priori. In the second case starting and finishing times become optimisation variables, with only duration specified for each water using operation. Consequently, this yields the true optimal schedule in terms of water use. Fixing the outlet concentration and the contaminant mass load for each water using operation in the absence of central reusable water storage initially renders the formulation a nonconvex mixed integer nonlinear program (MINLP) which is linearized exactly to yield a convex mixed integer linear program (MILP). On the other hand, allowing the outlet concentration to vary within predefined bounds while fixing both the water requirement and contaminant mass load yields an MINLP formulation for which global optimality cannot be guaranteed in complex problems. The developed formulation is applied to a published literature example as well as case studies involving production scheduling.

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_4,

4.1 Problem Statement

In a situation where the plant schedule already exists, the problem addressed in this chapter can be stated as follows. For each water using operation, given:

(i) the contaminant mass load,
(ii) the water requirement,
(iii) the starting and finishing times to achieve the desired effect, e.g. mass transfer, degree of cleanliness of the vessel, etc.,
(iv) maximum reusable water storage and
(v) maximum inlet and outlet concentrations,

determine the minimum amount of wastewater that can be achieved through the exploitation of reuse and recycle opportunities. *Reuse* refers to the use of an outlet water stream from operation j in another operation j', whereas *recycle* refers to the use of an outlet water stream from operation j in the same operation j. It is worthy of note that wastewater minimization is concomitant with reduction in freshwater intake.

In a situation where scheduling is not given beforehand, time is treated as an optimisation variable to provide a truly optimal schedule for freshwater demand and wastewater generation. The problem, in this particular case, can be stated as follows. Given the aforementioned (i)–(v) conditions for each water using operation as well as:

(vi) the production recipe for each product, including mean processing times in each unit operation,
(vii) the available units and their capacities,
(viii) the maximum storage capacity for each material and
(ix) the time horizon of interest.

determine the optimal production schedule which achieves minimum wastewater production through reuse and recycle. Embedding wastewater minimization within the scheduling framework implies that the starting and ending times need not be specified as these are derived directly from the optimal schedule. This is actually a true representation of a practical situation.

4.2 Problem Superstructure

Figures 4.1 and 4.2 depict the superstructures on which the mathematical model is based. Figure 4.1 represents a situation where reusable water storage does not exist. In this situation, water used in each water using operation j can be supplied from the fresh water header, the recycle/reuse water header or a combination of both headers. Water from each operation j can be recycled to the same operation, reused in downstream processes and/or dispensed with as effluent. On the other

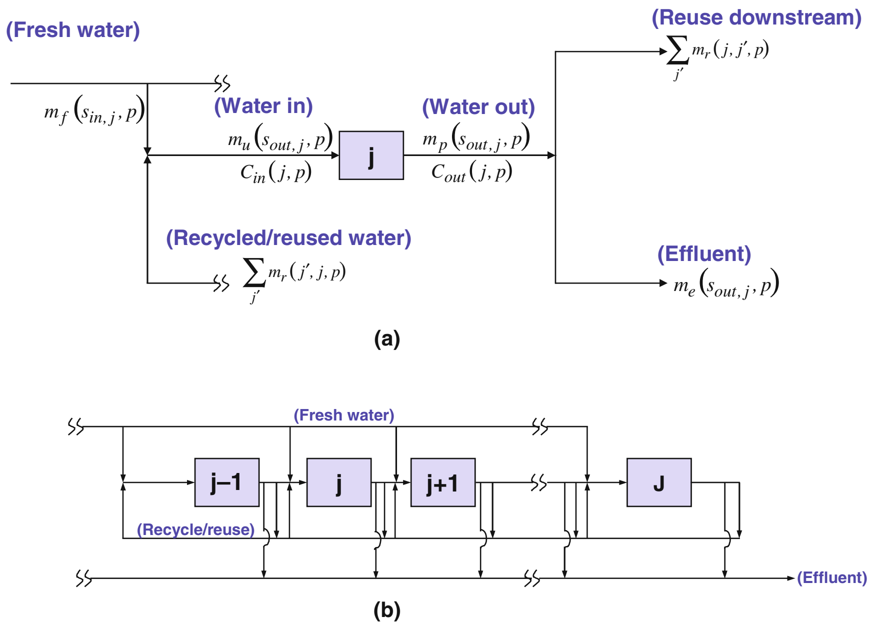


Fig. 4.1 Superstructure for the mathematical formulation with no reusable water storage (Majozi, 2005)

hand, Fig. 4.2 represents a situation where the reusable water storage exists. This situation provides water from reusable water storage as an additional source and water to reusable water storage as an additional sink for each operation *j*. Only water streams are shown in both superstructures for clarity purposes. The notation used in each superstructure follows that developed by Majozi and Zhu (2001) and applied by Majozi (2005). It is worthy of note that all the variables shown in the superstructure are dependent on time, which is represented by the index *p*.

Figures 4.1a and 4.2b show an exploded and condensed view of the superstructure, respectively. Each of the water using operations shown in the superstructure forms part of a complete batch chemical process. Different water using operations in the superstructure could belong to the same or distinct processes. The other process units in a complete batch plant are deliberately omitted from the diagram, since the focus of the mathematical formulation is only on water operations. However, operation and performance of the other units, which have a strong bearing on the starting and finishing times of water operations is captured by the scheduling module of the overall mathematical formulation. As mentioned earlier, this issue cannot be ignored, unless the starting and finishing times of the water operations are specified a priori, which is an oversimplification with significant practical implications. In practice, the operation of a batch water using operation is governed by an overall plant schedule, which also has to adhere to a predefined production plan. Therefore, it is almost impractical to treat any given water operation within a complete plant as an isolated operation.

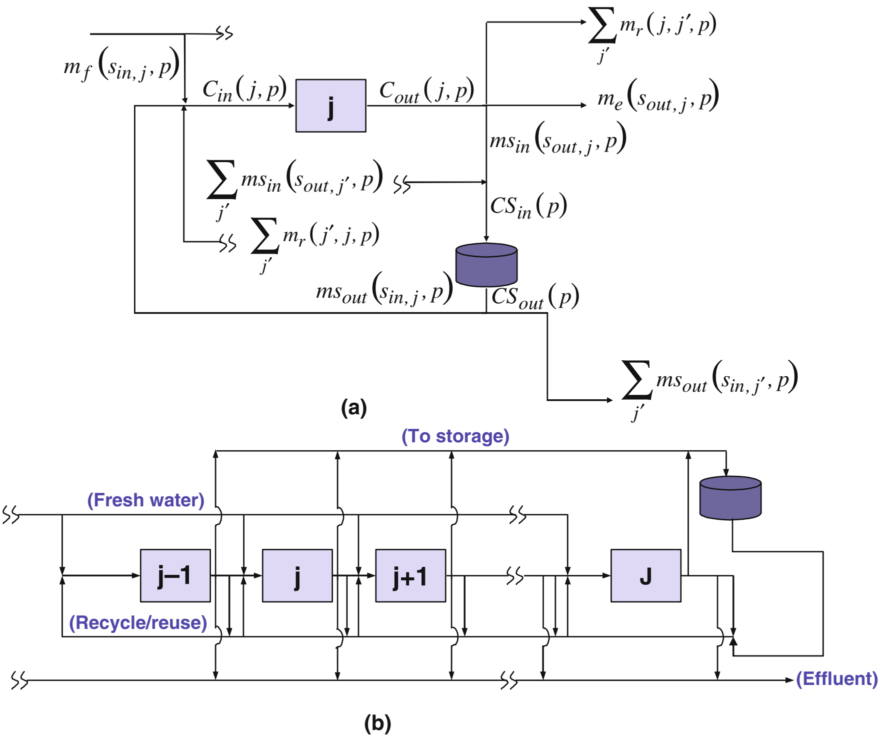


Fig. 4.2 Superstructure for the mathematical formulation with reusable water storage (Majozi, 2005)

4.3 Mathematical Model

The mathematical model presented in this section is based on the superstructures given in Figs. 4.1 and 4.2 and is made up of the following sets, variables and parameters.

Sets

S	$\{s \mid s$ is a state$\} = S_{\text{in},j} \cup S_{\text{out},j}$
J	$\{j \mid j$ is a unit$\}$
P	$\{p \mid p$ is a time point$\}$
$S_{\text{in},j}$	$\{s_{\text{in},j} \mid s_{\text{in},j}$ is an input state to unit j$\}$
$S_{\text{out},j}$	$\{s_{\text{out},j} \mid s_{\text{out},j}$ is an output state from unit j$\}$

Variables

$C_{\text{out}}(j,p)$	outlet concentration from unit j at time point p
$C_{\text{in}}(j,p)$	inlet concentration to unit j at time point p
$CS_{\text{out}}(p)$	outlet concentration from storage at time point p
$CS_{\text{in}}(p)$	inlet concentration to storage at time point p
$d(s,p)$	amount of state s delivered to customers at time point p, $s \in S_{\text{out},j}$
$m_e(s,p)$	amount of state s dispensed with as effluent at time point p, $s \in S_{\text{out},j}$
$m_f(s,p)$	amount of fresh water used in unit j at time point p, $s \in S_{\text{in},j}$
$m_p(s,p)$	amount of state s produced at time point p, $s \in S_{\text{out},j}$
$m_u(s,p)$	amount of state s used at time point p, $s \in S_{\text{in},j}$
$m_r(j,j',p)$	amount of water recycled or reused between two units j and j' at time point p
$ms_{\text{in}}(s,p)$	amount of state s that is transferred to storage at time point p, $s \in S_{\text{out},j}$
$ms_{\text{out}}(s,p)$	amount of state s that is transferred from storage to a particular unit j at time point p, $s \in S_{\text{in},j}$
$q_s(p)$	amount of water stored at time point p
$t_p(s,p)$	time at which state s is produced at time point p, $s \in S_{\text{out},j}$
$t_r(j,j',p)$	time at which water is recycled or reused between two units j and j' at time point p
$t_u(s,p)$	time at which state s is used at time point p, $s \in S_{\text{in},j}$
$ts_{\text{in}}(s,p)$	time at which state s is transferred to storage from operation j at time point p, $s \in S_{\text{out},j}$
$ts_{\text{out}}(s,p)$	time at which state s is transferred to operation j from storage at time point p, $s \in S_{\text{in},j}$
$y(s,p)$	binary variable associated with usage of state s at time point p, $s \in S_{\text{in},j}$
$y(j,j',p)$	binary variable associated with recycle or reuse between two units j and j' at time point p
$ys_{\text{in}}(s,p)$	binary variable associated with the transfer of state s from operation j to storage at time point p, $s \in S_{\text{out},j}$
$ys_{\text{out}}(s,p)$	binary variable associated with the transfer of state s from storage to operation j at time point p, $s \in S_{\text{in},j}$

Parameters

$C^U_{\text{out}}(j)$	maximum outlet contaminant concentration from unit j
$C^U_{\text{in}}(j)$	maximum inlet contaminant concentration to unit j
CE	treatment cost for effluent
$CR(j)$	cost of raw material associated with operation j

H	time horizon of interest
$M(j)$	mass-load of contaminant in unit j
$W^U(j)$	limiting/maximum water requirement in unit j
N	number of permitted stream splits
$\psi(j)$	mass-ratio between raw material stream and freshwater in unit j
$Q_s^0(s)$	initial amount of state s stored
Q_s^U	maximum capacity of reusable water storage
$SP(j)$	selling price for the product associated with operation j
V_j	capacity of a particular unit j
$W(j)$	fixed water requirement in unit j

The overall model is made up of two modules that are built within the same framework. One module focuses on the exploration of water reuse/recycle opportunities and the other on proper sequencing to capture the time dimension. To facilitate understanding, these modules are presented separately in the following sections.

4.3.1 Water Reuse/Recycle Module

In exploring the recycle and reuse opportunities within a complete batch process, four scenarios are mathematically formulated in the following sections. The first scenario is based on a fixed outlet concentration from each water using operation without the existence of reusable water storage. This situation allows for the quantity of water used in the operation to vary from the limiting water requirement. The limiting water requirement is the amount of water required if the initial contaminant concentration in a water using operation corresponds to the maximum permissible concentration. This is illustrated in Fig. 4.3. The second scenario is based on fixed water requirement for each water using operation without reusable water storage. The third and fourth scenarios consider the existence of reusable water storage and correspond to the first and second scenarios, respectively.

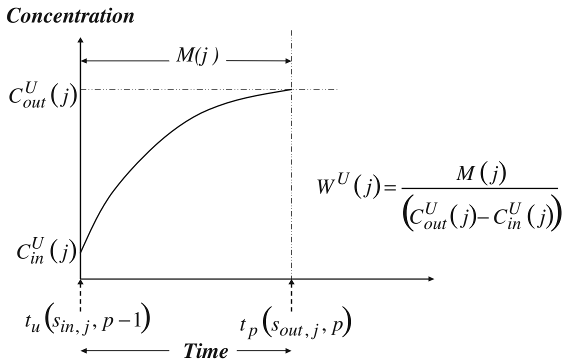


Fig. 4.3 Definition of the limiting water requirement (Majozi, 2005)

Scenario 1: *Formulation for fixed outlet concentration without reusable water storage*

This formulation is based on a superstructure given in Fig. 4.1.

$$m_u\left(s_{\text{in},j},p\right) = \sum_{j' \in J} m_r\left(j',j,p\right) + m_f\left(s_{\text{in},j},p\right), \quad \forall p \in P, s_{\text{in},j} \in S_{\text{in},j} \tag{4.1}$$

$$m_p\left(s_{\text{out},j},p\right) = m_e\left(s_{\text{out},j},p\right) + \sum_{j' \in J} m_r\left(j,j',p\right), \quad \forall p \in P, s_{\text{out},j} \in S_{\text{out},j} \tag{4.2}$$

$$\begin{aligned} m_p\left(s_{\text{out},j},p\right) C_{\text{out}}(j,p) &= m_u\left(s_{\text{in},j},p-1\right) C_{\text{in}}(j,p-1) \\ &\quad + M(j) y\left(s_{\text{in},j},p-1\right), \\ \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} &\in S_{\text{out},j}, p \in P, p > p_1 \end{aligned} \tag{4.3}$$

$$C_{\text{in}}(j,p) = \frac{\sum_{j'} m_r\left(j',j,p\right) C_{\text{out}}(j',p)}{\sum_{j'} m_r\left(j',j,p\right) + m_f\left(s_{\text{in},j},p\right)}, \forall j,j' \in J, p \in P, s_{\text{in},j} \in S_{\text{in},j} \tag{4.4}$$

$$C_{\text{out}}(j,p) = C_{\text{out}}^{U}(j) y\left(s_{\text{in},j},p-1\right), \forall j \in J, p \in P, p > p_1, s_{\text{in},j} \in S_{\text{in},j} \tag{4.5}$$

$$m_u\left(s_{\text{in},j},p\right) \le W^U(j)\, y\left(s_{\text{in},j},p\right), \forall s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{4.6}$$

Constraints (4.1) states that the inlet stream into any operation j comprises of the reuse/recycle streams from all water using operations j' as well as the fresh water stream. As mentioned early on in this chapter, *reuse* refers to the use of an outlet water stream from operation j in another operation j', whereas *recycle* refers to the use of an outlet water stream from operation j in the same operation j. Constraint (4.2) states that the outlet stream from any operation j can be reused in other processes j', recycled to the same process j and/or dispensed with as effluent. Constraint (4.3) is the mass balance around unit j. It states that the contaminant mass-load difference between outlet and inlet streams for the same unit j is the contaminant mass-load picked up in unit j. Attention should also be brought to the fact that the inlet and outlet streams of the same unit j correspond to consecutive time points $p-1$ and p. Constraint (4.4) is the definition of the inlet contaminant concentration to unit j. It is defined as the ratio of the overall mass-load in recycle/reuse streams to the total amount of the inlet stream. Constraint (4.5) states that the outlet concentration from any unit j is fixed at a maximum predefined concentration corresponding to the same unit. It should be noted that streams are expressed in quantities instead of flowrates, which is indicative of any batch operation. The total quantity of water used at any point in time cannot exceed the limiting amount as stated in constraint (4.6).

Constraints (4.1), (4.2), (4.3), (4.4), (4.5) and (4.6) constitute a nonconvex nonlinear model due to constraints (4.3) and (4.4), which involve bilinear terms. Nonconvexity, and not necessarily nonlinearity, is a disadvantageous feature in any model, since global optimality cannot be guaranteed. Therefore, if can be avoided, it should. This is achieved by either linearizing the model or using convexification techniques where applicable. In this instance, the first option was proven possible as shown below.

Observing that the amount of water in a given process is conserved, constraints (4.3) can be written as:

$$C_{\mathrm{in}}(j,p-1) = C_{\mathrm{out}}(j,p) - \frac{M(j)\,y\left(s_{\mathrm{in},j},p-1\right)}{m_u\left(s_{\mathrm{in},j},p-1\right)}, \quad \forall j \in J, s_{\mathrm{in},j} \in S_{\mathrm{in},j}, p \in P, p > p_1 \tag{4.7}$$

Also, constraints (4.4) can be written as:

$$C_{\mathrm{in}}(j,p-1) = \frac{\sum_{j'} m_r\left(j',j,p-1\right) C_{\mathrm{out}}\left(j',p-1\right)}{m_u\left(s_{\mathrm{in},j},p-1\right)}, \quad \forall j,j' \in J, p \in P, p > p_1, s_{\mathrm{in},j} \in S_{\mathrm{in},j} \tag{4.8}$$

Then, substituting constraints (4.7) in (4.8) leads to:

$$C_{\mathrm{out}}(j,p)\,m_u\left(s_{\mathrm{in},j},p-1\right) - M(j)\,y\left(s_{\mathrm{in},j},p-1\right) = \sum_{j'} m_r\left(j',j,p-1\right) C_{\mathrm{out}}\left(j',p-1\right) \quad \forall j,j' \in J, s_{\mathrm{in},j} \in S_{\mathrm{in},j}, p \in P, p > p_1 \tag{4.9}$$

Since the outlet concentration is fixed according to constraints (4.5), constraints (4.9) can be written as follows:

$$C_{\mathrm{out}}^{U}(j)y\left(s_{\mathrm{in},j},p-1\right)m_u\left(s_{\mathrm{in},j},p-1\right) - M(j)y\left(s_{\mathrm{in},j},p-1\right) = \sum_{j'} m_r\left(j',j,p-1\right) y\left(s_{\mathrm{in},j'},p-2\right) C_{\mathrm{out}^U}\left(j'\right) \quad \forall j,j' \in J, s_{\mathrm{in},j} \in S_{\mathrm{in},j}, p \in P, p > p_2 \tag{4.10}$$

Constraints (4.10) still entails nonconvex bilinear terms comprising of a binary and a continuous variable. However, this type of bilinearity can be readily linearized exactly using Glover transformation (1975). Constraints (4.11), (4.12), (4.13), (4.14) and (4.15) together constitute a linearized form of constraints (4.10). In constraints (4.11), the first and the second bilinear terms from constraints (4.10) have been replaced by continuous variables Γ_1 and Γ_2, respectively. Γ_1 is defined in constraints (4.12) and (4.13), and Γ_2 in constraints (4.14) and (4.15).

$$C_{\text{out}}^{U}(j)\,\Gamma_1\left(s_{\text{in},j},p\right) - M(j)\,y\left(s_{\text{in},j},p-1\right) = \sum_{j'} \Gamma_2\left(j',j,p\right) C_{\text{out}^U}\left(j'\right), \quad \forall j,j' \in J, s_{\text{in},j} \in S_{\text{in},j} \tag{4.11}$$

$$m_u\left(s_{\text{in},j},p-1\right) - W^U(j)\left(1 - y\left(s_{\text{in},j},p-1\right)\right) \le \Gamma_1\left(s_{\text{in},j},p\right) \le m_u\left(s_{\text{in},j},p-1\right), \quad \forall s_{\text{in},j} \in S_{\text{in},j}, p \in P, p > p_1, j \in J \tag{4.12}$$

$$\Gamma_1\left(s_{\text{in},j},p\right) \le W^U(j)\,y\left(s_{\text{in},j},p-1\right), \forall s_{\text{in},j} \in S_{\text{in},j}, p \in P, p > p_1 \tag{4.13}$$

$$m_r\left(j',j,p-1\right) - W^U(j)\left(1 - y\left(s_{\text{in},j'},p-2\right)\right) \le \Gamma_2\left(j',j,p\right) \le m_r\left(j',j,p-1\right), \quad \forall s_{\text{in},j'} \in S_{\text{in},j'}, p \in P, p > p_2, j, j' \in J \tag{4.14}$$

$$\Gamma_2\left(j',j,p\right) \le W^U(j)\,y\left(s_{\text{in},j'},p-2\right), \quad \forall s_{\text{in},j'} \in S_{\text{in},j}, \forall j,j' \in J, p \in P, p > p_2 \tag{4.15}$$

Constraints (4.1), (4.2), (4.5), (4.6), (4.11), (4.12), (4.13), (4.14) and (4.15) constitute a full model for scenario 1, excluding the sequence constraints which are presented in Section 4.3.2. It is evident from all the constraints that this model is a mixed integer linear program (MILP) for which global optimality is guaranteed. It is worth noting, however, that ensuring linearity of the model entailed a penalty of two additional variables and four more constraints. In certain instances, this increase in problem size could result in longer CPU times. However, in all the instances tested using this model, ensuring linearity did not adversely impact on CPU time, which could be ascribed to the structure of the overall model.

Scenario 2: *Formulation for fixed water quantity without reusable water storage*

The following formulation, which is also based on the superstructure given in Fig. 4.1, is applicable in a situation where the quantity of water is fixed and the outlet concentration is allowed to vary. In this situation, constraints (4.1), (4.2), (4.3) and (4.4) still hold, but constraints (4.5) and (4.6) have to be modified as follows.

$$C_{\text{out}}(j,p) \le C_{\text{out}}^{U}(j)\,y\left(s_{\text{in},j},p-1\right), \quad \forall j \in J, p \in P, p > p_1, s_{\text{in},j} \in S_{\text{in},j} \tag{4.16}$$

$$m_u\left(s_{\text{in},j},p\right) = W^U(j)\,y\left(s_{\text{in},j},p\right), \forall s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{4.17}$$

Unfortunately, in this situation there is no opportunity for exact linearization due to constraints (4.8). The opportunity for *exact* linearization arises when a bilinear term involves a continuous variable and a binary variable. Only scenario 1 can be reduced to this structure, hence the linearization. Therefore, the overall model comprising of constraints (4.1), (4.2), (4.3), (4.4), (4.16) and (4.17) is a nonconvex MINLP, which implies that global optimality cannot be guaranteed even if concomitant with significant improvements from the current operation.

Scenario 3: *Formulation for fixed outlet concentration with reusable water storage*

The formulation presented in this section corresponds to the superstructure shown in Fig. 4.2 in which central reusable water storage is taken into consideration. In this situation, constraints (4.3), (4.5) and (4.6) still hold, but constraints (4.1), (4.2) and (4.4) have to be modified as follows.

$$m_u\left(s_{\text{in},j},p\right) = \sum_{j'} m_r\left(j',j,p\right) + m_f\left(s_{\text{in},j},p\right) + ms_{\text{out}}\left(s_{\text{in},j},p\right), \quad \forall j,j' \in J, p \in P, s_{\text{in},j} \in S_{\text{in},j} \tag{4.18}$$

$$m_p\left(s_{\text{out},j},p\right) = m_e\left(s_{\text{out},j},p\right) + \sum_{j'} m_r\left(j,j',p\right) + ms_{\text{in}}\left(s_{\text{out},j},p\right), \quad \forall j,j' \in J, p \in P, s_{\text{out},j} \in S_{\text{out},j} \tag{4.19}$$

$$C_{\text{in}}\left(j,p\right) = \frac{\sum_{j'} m_r\left(j',j,p\right) C_{\text{out}}(j',p) + ms_{\text{out}}\left(s_{\text{in},j},p\right) CS_{\text{out}}\left(p\right)}{\sum_{j'} m_r\left(j',j,p\right) + m_f\left(s_{\text{in},j},p\right) + ms_{\text{out}}\left(s_{\text{in},j},p\right)}, \quad \forall j,j' \in J, p \in P, s_{\text{in},j} \in S_{\text{in},j} \tag{4.20}$$

Constraints (4.18) states that the inlet stream into any operation j is made up of recycle/reuse stream, fresh water stream and a stream from reusable water storage. On the other hand, the outlet stream from operation j can be dispensed with as effluent, reused in other processes, recycled to the same operation and/or sent to reusable water storage as shown in constraints (4.19). The inlet concentration into operation j is the ratio of the contaminant amount in the inlet stream and the quantity of the inlet stream as stated in constraints (4.20). The amount of contaminant in the inlet stream to operation j consists of the contaminant in the recycle/reuse stream and the contaminant in the reusable water storage stream. The following storage specific constraints are also imperative for the completeness of the model for scenario 3.

$$qs\left(p\right) = qs\left(p-1\right) + \sum_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j},p\right) - \sum_{s_{\text{in},j}} ms_{\text{out}}\left(s_{\text{in},j},p\right), \quad \forall j \in J, p \in P, p > p_1, s_{\text{out},j} \in S_{\text{out},j}, s_{\text{in},j} \in S_{\text{in},j} \tag{4.21}$$

$$qs\left(p_1\right) = Q_s^0 - \sum_{s_{\text{in},j}} ms_{\text{out}}\left(s_{\text{in},j},p_1\right), \ \forall j \in J, s_{\text{in},j} \in S_{\text{in},j} \tag{4.22}$$

$$qs\left(p\right) \leq Q_s^U, \ \forall p \in P \tag{4.23}$$

$$CS_{\text{in}}(p) = \frac{\sum\limits_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j}\right) C_{\text{out}}(j,p)}{\sum\limits_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j},p\right)}, \quad \forall j \in J, p \in P, s_{\text{out},j} \in S_{\text{out},j} \tag{4.24}$$

$$CS_{\text{out}}(p) = \frac{qs(p-1)CS_{\text{out}}(p-1) + \sum\limits_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j}\right) C_{\text{out}}(j,p)}{qs(p-1) + \sum\limits_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j},p\right)}, \quad \forall j \in J, p \in P, p > p_1, s_{\text{out},j} \in S_{\text{out},j} \tag{4.25}$$

$$CS_{\text{out}}(p_1) = CS_{\text{out}}^0 \tag{4.26}$$

Constraints (4.21) is the mass balance around reusable water storage tank. It states that the amount stored at any time point p is determined by the amount stored at the previous time point $p-1$ plus the difference between the quantity transferred from and the quantity transferred to the water using operations at time point p. However, at the beginning of the time horizon of interest, none of the water using operations is complete and ready to transfer to storage. Also, the amount stored at the previous time point corresponds to initial amount of reusable water available in storage Q_s^0. Therefore, at the beginning of the time horizon, constraints (4.22) replaces constraints (4.21). Constraints (4.23) ensures that the amount of water stored at any point in time does not exceed the capacity of reusable water storage.

Constraints (4.24) and (4.25) respectively give the inlet and outlet concentrations for the reusable water storage tank. At any given time point p, the inlet concentration is the ratio of the contaminant load in all streams transferred from water using operations to the overall quantity of the stream transferred to reusable water storage tank. The outlet concentration at any time point p is defined as the contaminant load at the previous time point $p-1$ plus the contaminant load in the incoming stream from water using operations divided by the total quantity of reusable water in the storage tank. This is also the definition of the contaminant concentration inside the reusable water storage tank. Hence, it is assumed that the outlet concentration from the storage tank is the same as the concentration inside the tank. This is indeed a valid assumption if perfect mixing is achieved within the tank. It is evident that constraints (4.25) is not applicable at the beginning of the time horizon of interest for reasons similar to constraints (4.21). Therefore, constraints (4.26) replaces constraints (4.25) at the beginning of the time horizon.

Constraints (4.18), (4.19), (4.3), (4.20), (4.5), (4.6), (4.21), (4.22), (4.23), (4.24), (4.25) and (4.26) constitute a complete water reuse/recycle mathematical model for scenario 3, which is also a nonconvex MINLP. Although the outlet concentration and contaminant mass load from water using operations is fixed as in scenario 1, the overall formulation cannot be linearized due to variability of outlet concentration from reusable water storage tank.

Scenario 4: *Formulation for fixed water quantity with reusable water storage*

Constraints (4.18), (4.19), (4.3), (4.20), (4.16), (4.17), (4.21), (4.22), (4.23), (4.24), (4.25) and (4.26) together constitute a complete water reuse/recycle model for a situation in which the quantity of water in each water using operation is fixed. This is also a nonconvex MINLP for which exact linearization is not possible.

4.3.2 Sequencing/Scheduling Module

In order to handle batch operations effectively, the time dimension cannot be ignored. This is due to the fact that almost all operations within the batch process environment are time dependent. Figure 4.4 shows comparison between batch and continuous processes on the exploration of water reuse opportunities. In continuous operations, only the concentration constraints determines the feasibility of water reuse from one process to another. This implies that if the outlet water concentration from process A is less than the maximum allowed inlet water concentration to process B, then water from process A could be reused in process B. On the other hand, if water from one process, say process B, is at a concentration higher than the maximum allowed in another process, say process A, the water reuse opportunity from process B to process A is nullified.

In batch operations, there is also a time constraints to be satisfied, in addition to the concentration constraints. This particular feature renders batch operations more challenging than their continuous counterparts. As shown in Fig. 4.4, even if the concentration constraints is obeyed, water from process A cannot be readily reused in process B if process A commences after process B. This constraints could be

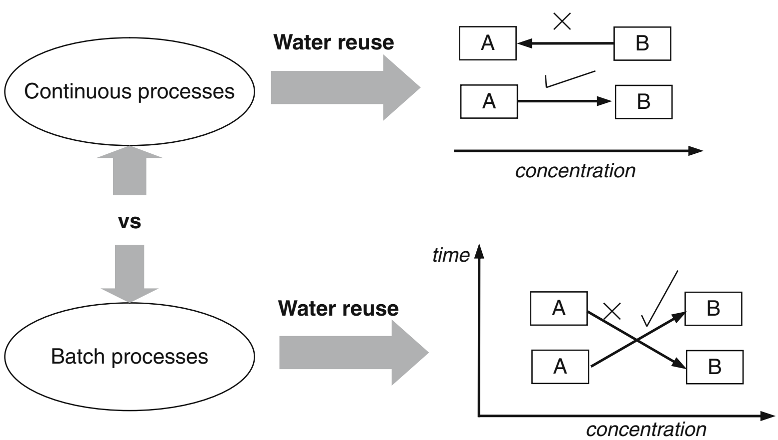


Fig. 4.4 Comparison of continuous and batch processes (Majozi, 2005)

bypassed by using reusable water storage tanks wherein water is stored for later use. However, this cannot be efficiently achieved without properly addressing the time constraints that is inherent in batch operations.

Sequencing in the Absence of Reusable Water Storage

The following constraints address the time dimension for water reuse/recycle in batch processes in the absence of central reusable water storage.

$$y_r\left(j,j',p\right) \le y\left(s_{\text{in},j'},p\right), \quad \forall s_{\text{in},j'} \in S_{\text{in},j'},p \in P,j,j' \in J \tag{4.27}$$

$$t_r\left(j,j',p\right) \le t_p\left(s_{\text{out},j},p\right) + H\left(1 - y_r\left(j,j',p\right)\right), \quad \forall j,j' \in J,s_{\text{out},j} \in S_{\text{out},j},p \in P \tag{4.28}$$

$$t_r\left(j,j',p\right) \ge t_p\left(s_{\text{out},j},p\right) - H\left(1 - y_r\left(j,j',p\right)\right), \quad \forall j,j' \in J,s_{\text{out},j} \in S_{\text{out},j},p \in P \tag{4.29}$$

$$t_r\left(j,j',p\right) \le t_u\left(s_{\text{in},j'},p\right) + H\left(1 - y_r\left(j,j',p\right)\right), \quad \forall j,j' \in J,s_{\text{in},j'} \in S_{\text{in},j'},p \in P \tag{4.30}$$

$$t_r\left(j,j',p\right) \ge t_u\left(s_{\text{in},j'},p\right) - H\left(1 - y_r\left(j,j',p\right)\right), \quad \forall j,j' \in J,s_{\text{in},j'} \in S_{\text{in},j'},p \in P \tag{4.31}$$

$$t_u\left(s_{\text{in},j},p\right) \ge t_p\left(s_{\text{out},j},p'\right) - H\left(2 - y\left(s_{\text{in},j},p\right) - y\left(s_{\text{in},j},p'-1\right)\right), \quad \forall j \in J,s_{\text{in},j} \in S_{\text{in},j},s_{\text{out},j} \in S_{\text{out},j},\forall p,p' \in P,p' > p_1,p \ge p' \tag{4.32}$$

$$t_u\left(s_{\text{in},j},p\right) \ge t_p\left(s_{\text{in},j},p'\right) - H\left(2 - y\left(s_{\text{in},j},p\right) - y\left(s_{\text{in},j},p'\right)\right), \quad \forall j \in J,s_{\text{in},j} \in S_{\text{in},j},p,p' \in P,p \ge p' \tag{4.33}$$

$$t_u\left(s_{\text{out},j},p\right) \ge t_p\left(s_{\text{out},j},p'\right) - H\left(2 - y\left(s_{\text{out},j},p\right) - y\left(s_{\text{out},j},p'\right)\right), \quad \forall j \in J,s_{\text{out},j} \in S_{\text{out},j},p,p' \in P,p \ge p' \tag{4.34}$$

Constraints (4.27) states that if water is recycled from operation j to operation j' at a given time point p, then operation j' should commence at time point p. However, the fact that operation j' commences at time point p does not necessarily mean that there is a corresponding recycle/reuse stream at time point p. This is due to the fact that operation j' could be using freshwater instead of recycle/reuse stream. Constraints (4.28) and (4.29) together ensure that water recycle/reuse from operation j to operation j' coincides with the completion of operation j at time point p. Similarly, constraints (4.30) and (4.31) ensure that water recycle/reuse from operation $j$to operation j' coincides with the start of operation j' at time point p. Constraints (4.32) states that any operation j will start after the previous task in the same operation j is complete at time point p. Constraints (4.33) and (4.34) respectively state that if an operation j starts or ends at two distinct time points, then the later time point must correspond to a later time. These constraints have proven to improve CPU time and ensure robustness and feasibility in the model.

Sequencing in the Presence of Reusable Water Storage

As mentioned early on, constraints (4.27), (4.28), (4.29), (4.30), (4.31), (4.32), (4.33) and (4.34) suffice in the absence of reusable water storage. However, in the presence of reusable water storage, the following additional sequence constraints are necessary.

$$ts_{\text{in}}\left(s_{\text{out},j},p\right) \geq t_p\left(s_{\text{out},j},p\right) - H\left(1 - ys_{\text{in}}\left(s_{\text{out},j},p\right)\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{4.35}$$

$$ts_{\text{in}}\left(s_{\text{out},j},p\right) \leq t_p\left(s_{\text{out},j},p\right) + H\left(1 - ys_{\text{in}}\left(s_{\text{out},j},p\right)\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{4.36}$$

$$ys_{\text{in}}\left(s_{\text{out},j},p\right) \leq y\left(s_{\text{in},j},p-1\right), \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j}, p \in P, p > p_1 \tag{4.37}$$

Constraints (4.35) and (4.36) stipulate that when the water stream is transferred from operation j to reusable water storage, then the time of transfer should coincide with the completion of operation j. however, operation j will only be completed and able to transfer water to storage at time point p if it started at time point $p-1$. Also, the fact that operation j commenced at time point $p-1$ does not necessarily mean that it will transfer water to storage at time point p, since this water could be immediately reused/recycled and/or dispensed with as effluent. This is captured by constraints (4.37). The following constraints (4.38), (4.39) and (4.40) are similar to (4.35), (4.36) and (4.37), but apply to the outlet stream of reusable water storage.

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \geq t_u\left(s_{\text{in},j},p\right) - H\left(1 - ys_{\text{out}}\left(s_{\text{in},j},p\right)\right), \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{4.38}$$

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \leq t_u\left(s_{\text{in},j},p\right) + H\left(1 - ys_{\text{out}}\left(s_{\text{in},j},p\right)\right), \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{4.39}$$

$$ys_{\text{out}}\left(s_{\text{in},j},p\right) \leq y\left(s_{\text{in},j},p\right), \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{4.40}$$

Constraints (4.38) and (4.39) state that when water stream is transferred from storage to any operation j for reuse, then the time of transfer must coincide with the start of operation j. Constraints (4.40) ensures that whenever a water stream is transferred from storage to operation j at time point p, then operation j must commence at time point p. However, operation j can start at time point p even if there is no reusable water stream transferred from storage, since water could be received from recycle/reuse and fresh water streams.

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \geq ts_{\text{out}}\left(s_{\text{in},j'},p'\right) - H\left(2 - ys_{\text{out}}\left(s_{\text{in},j},p\right) - ys_{\text{out}}\left(s_{\text{in},j'},p'\right)\right), \quad \forall j,j' \in J, s_{\text{in},j}, s_{\text{in},j'} \in S_{\text{in},j}, p,p' \in P, p \geq p' \tag{4.41}$$

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \geq ts_{\text{out}}\left(s_{\text{in},j'},p\right) - H\left(2 - ys_{\text{out}}\left(s_{\text{in},j},p\right) - ys_{\text{out}}\left(s_{\text{in},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{in},j}, s_{\text{in},j'} \in S_{\text{in},j}, p \in P \tag{4.42}$$

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \leq ts_{\text{out}}\left(s_{\text{in},j'},p\right) + H\left(2 - ys_{\text{out}}\left(s_{\text{in},j},p\right) - ys_{\text{out}}\left(s_{\text{in},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{in},j}, s_{\text{in},j'} \in S_{\text{in},j}, p \in P \tag{4.43}$$

Constraints (4.41) ensures that if reusable water is transferred from reusable water storage to operation j' at time point p' and later transferred to the same or another operation j at time point p, then the later time point must correspond to a later time. If the transfer of water from reusable water storage to different operations j and j' takes place at the same time point p, then this time point must correspond to exactly the same time as enforced by both constraints (4.42) and (4.43).

$$ts_{\text{in}}\left(s_{\text{out},j},p\right) \geq ts_{\text{in}}\left(s_{\text{out},j'},p'\right) - H\left(2 - ys_{\text{in}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p'\right)\right), \quad \forall j,j' \in J, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p,p' \in P, p \geq p' \tag{4.44}$$

$$ts_{\text{in}}\left(s_{\text{out},j},p\right) \geq ts_{\text{in}}\left(s_{\text{out},j'},p\right) - H\left(2 - ys_{\text{in}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p \in P \tag{4.45}$$

$$ts_{\text{in}}\left(s_{\text{out},j},p\right) \leq ts_{\text{in}}\left(s_{\text{out},j'},p\right) + H\left(2 - ys_{\text{in}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p \in P \tag{4.46}$$

Constraints (4.44), (4.45) and (4.46) are similar to constraints (4.41), (4.42) and (4.43), but apply to the inlet stream of reusable water storage.

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \geq ts_{\text{in}}\left(s_{\text{out},j'},p'\right) - H\left(2 - ys_{\text{out}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p'\right)\right), \quad \forall j,j' \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p,p' \in P, p \geq p' \tag{4.47}$$

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \geq ts_{\text{in}}\left(s_{\text{out},j'},p\right) - H\left(2 - ys_{\text{out}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p \in P \tag{4.48}$$

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \leq ts_{\text{in}}\left(s_{\text{out},j'},p\right) + H\left(2 - ys_{\text{out}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p \in P \tag{4.49}$$

Constraints (4.47) states that if reusable water is transferred from operation j' to storage at time point p' and later transferred from reusable water storage to the same or another operation j at time point p, then the latter transfer must correspond to a later time. On the other hand, if the transfer of reusable water into and out of reusable water storage occurs at the same time point p, then this should correspond to the same time, i.e. the time of transfer from operation j' to reusable water storage must be equal to the time of transfer from reusable water storage to operation j. This is captured by constraints (4.48) and (4.49).

It is worthy of note that the water reuse/recycle and sequencing modules presented in Sections 4.3.1 and 4.3.2 treat water using operations as if they were isolated from the rest of the batch process. This implies that the impact of the other units in the batch process on the exploration of water reuse/recycle opportunities is ignored. Moreover, any water using operation j is treated as a dedicated operation, which implies that only water using tasks are conducted in any given operation j. For example, operation j cannot operate as both a reactor and a liquid–liquid extraction unit. However, the scenario in which the impact of other units on water using operations within the overall batch process is taken into consideration and a scenario in which a given operation or unit conducts more than one task can be readily embedded within a broader scheduling framework. This framework, which entails stoichiometric, mass balance, storage, duration, assignment/preemptive and sequence constraints, has been presented in detail in Chapter 2, hence will not be detailed in this chapter to avoid repetition. The aforementioned two scenarios are handled by sequence and assignment/preemptive constraints, respectively. The sequence constraints ensure that, given the duration of operation j, the starting times over a given time horizon are dependent on the starting and finishing times of the surrounding operations within the batch process. This implies that only duration and not starting times need to be specified during problem specification. The assignment or preemptive constraints are necessary if a given operation or unit j is capable of conducting more than one task. These ensure that, at any given point in time, operation j conducts at most one task.

4.4 Objective Function

In addition to their ability to capture the multidimensionality of batch operations, another advantage of mathematical programming techniques is the flexibility and adaptability of the performance index, i.e. the objective function. In a design problem, the objective function can take a form of a capital cost investment function. In a scheduling problem it can be minimization of makespan, maximization of throughput, maximization of revenue, etc. In this chapter, the objective function will either be the minimization of freshwater requirement, which is similar to minimisation of wastewater generation or the maximization of profit with emphasis on operating costs, as shown in the following literature example and case studies.

4.5 Literature Example

The data for the literature example (Wang and Smith, 1995) is shown in Table 4.1. The literature example involves three water using operations for which the limiting concentrations and water quantities are specified. The starting and finishing times are also given a priori, since the optimization of the overall plant schedule is ignored. The quantity of water required in each operation is calculated from flowrate and duration. Therefore, the quantities of water required for operations 1, 2 and 3 are 100, 40 and 25 t, respectively. This implies that the total freshwater requirement without exploiting recycle/reuse opportunities is 165 t over a 1.5 h time horizon. The objective is to minimise freshwater requirement and effluent production over the same time horizon.

Table 4.1 Data for the literature example (Majozi, 2005)

Operation (j)	Flowrate (t/h)	$C^U_{\text{out}}(j)$ (kg/t)	$C^U_{\text{in}}(j)$ (kg/t)	$t_u(s,p)$ (h)	$t_p(s,p)$ (h)	$M(j)$ (kg)
1	100	0.4	0.1	0.5	1.5	30
2	80	0.2	0	0	0.5	8
3	50	0.2	0.1	0.5	1.0	2.5

Objective Function

The objective of the formulation is to minimize the amount of freshwater required, hence the amount of wastewater generated, over the 1.5 h time horizon as shown below.

$$\text{Minimize} \quad \sum_{p_1}^{p_3} \left(\sum_{s_{\text{in},1}}^{s_{\text{in},3}} m_f\left(s_{\text{in},j},p\right) \right) \tag{4.50}$$

Computational Results

All the results presented in this section were obtained using different GAMS solvers in a 1.82 GHz Pentium 4 processor. The results presented in this chapter were not compared with those obtained by Wang and Smith (1995), since the latter method was proven to be more appropriate for semi-continuous rather than completely batch operations (Majozi et al., 2006).

Scenario 1: Formulation for fixed outlet concentration without reusable water storage

The results for this scenario were obtained using GAMS 2.5/CPLEX. The overall mathematical formulation entails 385 constraints, 175 continuous variables and 36 binary/discrete variables. Only 4 nodes were explored in the branch and bound algorithm leading to an optimal value of 215 t (fresh- and waste-water) in 0.17 CPU seconds. Figure 4.5 shows the water reuse/recycle network corresponding to fixed outlet concentration and variable water quantity for the literature example. It is worth noting that the quantity of water to processes 1 and 3 has been reduced by 5 and 12.5 t, respectively, from the specified quantity in order to maintain the outlet concentration at the maximum level. The overall water requirement has been reduced by almost 35% from the initial amount of 165 t.

Scenario 2: Formulation for fixed water quantity without reusable water storage

The results for scenario 2 were obtained using GAMS 2.5/DICOPT. The NLP and MILP combination of solvers selected for DICOPT were MINOS5 and CPLEX, respectively. The overall formulation involves 421 constraints, 175 continuous variables and 36 discrete variables. Only 2 nodes were explored in the branch and

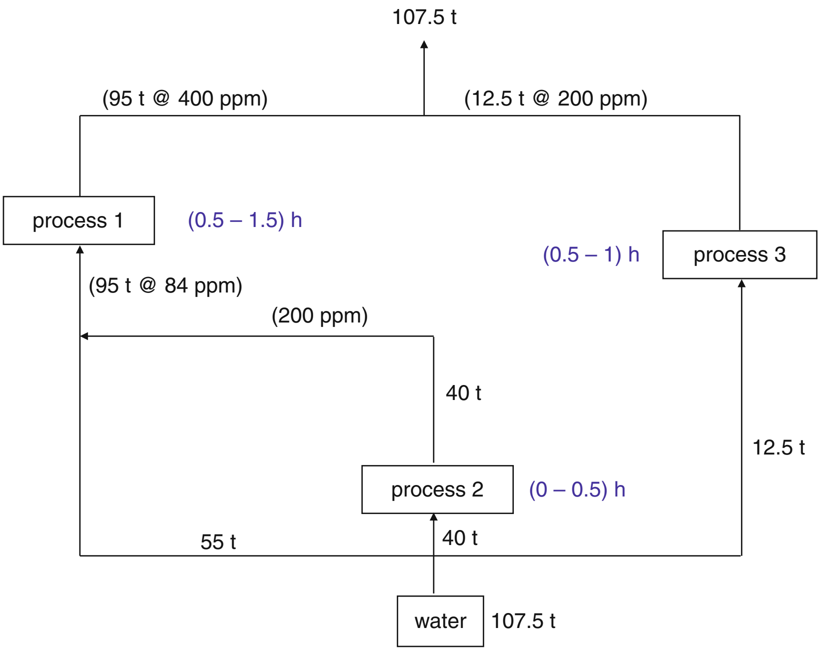


Fig. 4.5 Water reuse/recycle network corresponding to fixed outlet concentration (Majozi, 2005)

bound tree and three major iterations between the MILP master problem and the NLP subproblem were necessary to get to the optimal solution. The objective value of 250 corresponds to freshwater requirement of 125 t and wastewater production of 125 t. The problem was solved in 0.63 CPU seconds. Figure 4.6 shows the corresponding water reuse/recycle network. It is worthy of note that the quantity of water into each of the processes is as specified in the problem table (Table 4.1). However, the exploitation of reuse/recycle opportunities allows water from process 2 to be reused in processes 1 and 3, thereby culminating in 25% reduction in freshwater requirement and wastewater production. It should also be noted that fixing water quantity allows outlet concentration to vary from the specified maximum.

It is worthy of mention that, whilst the water requirement could be optimal, the concomitant water network might not be optimal from the operational and economic standpoint. For example, in Fig. 4.6 a splitter of a fixed ratio or a flowmeter would have to be installed at the exit of process 2 to ensure that the correct amount of reusable water is transferred to processes 1 and 3. An alternative, simpler design corresponding to the same freshwater requirement is shown in Fig. 4.7. The design engineer should, therefore, be aware that for a given optimal freshwater requirement and/or wastewater production there could be more than one water reuse/recycle networks. The following additional constraints could be added to the model to cater for the splits. In constraints (4.51), Nstands for the number of permitted stream splits.

$$\sum_{j'} y_r\left(j,j',p\right) \leq N+1, \quad \forall j,j' \in J,p \in P \tag{4.51}$$

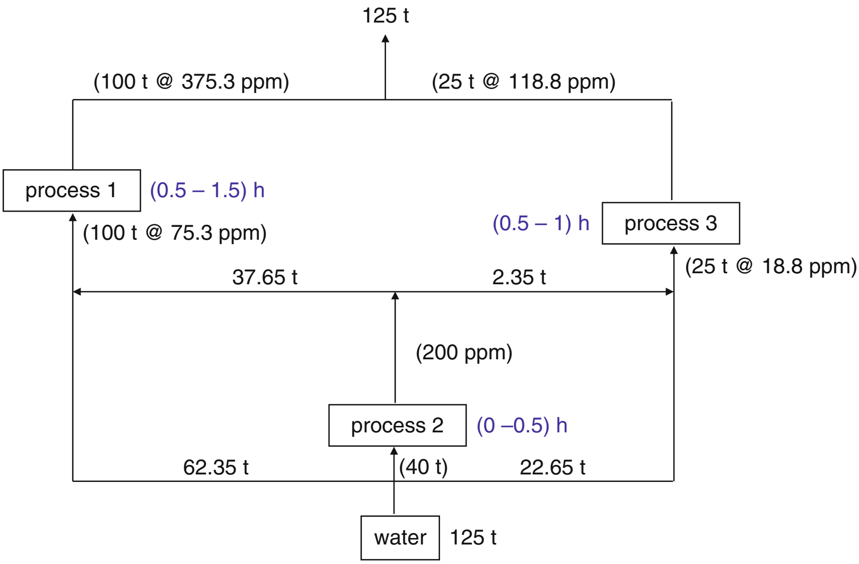


Fig. 4.6 Water reuse/recycle network corresponding to fixed water quantity (Majozi, 2005)

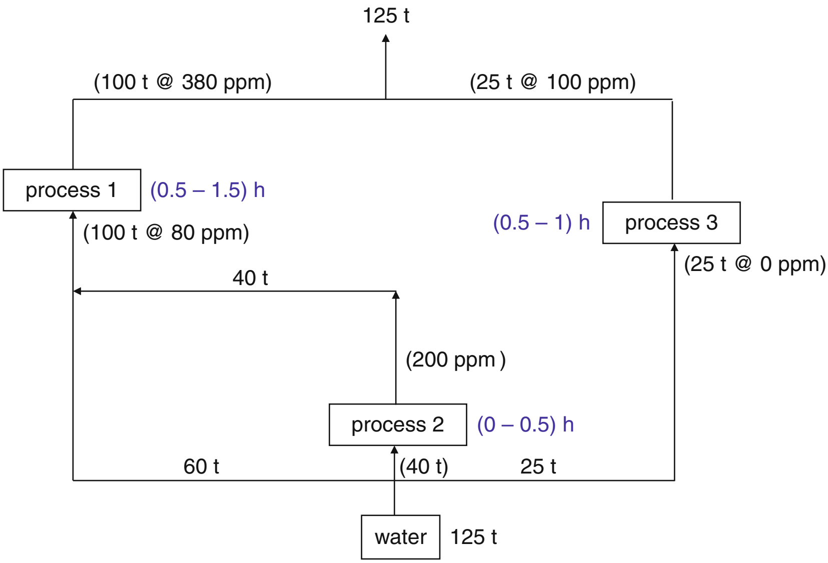


Fig. 4.7 Alternative water reuse/recycle network corresponding to fixed water quantity (Majozi, 2005)

For this particular example, scenarios 3 and 4 yielded the same networks as scenarios 1 and 2, respectively. This is due to the fact that processes 1 and 3 commence as soon as process 2 is complete, i.e. at 0.5 h. As a result there exists no opportunity for reusable water storage.

4.6 First Case Study

The data for the case study taken from an agrochemical manufacturing facility is given in Table 4.2. The corresponding Gantt chart is shown in Fig. 4.8. The processes given in Table 4.2 were specifically selected on the basis of a common

Table 4.2 Data for the first case study

Operation (j)	Water (kg)	C^U_{out} (j) (kg salt/kg water)	C^U_{in} (j) (kg salt/kg water)	$t_u(s,p)$ (h)	$t_p(s,p)$ (h)	$M(j)$ (kg)
A	1000	0.1	0	0	3	100
B	280	0.51	0.25	0	4	72.8
C	400	0.1	0.1	4	5.5	0
D	280	0.51	0.25	2	6	72.8
E	400	0.1	0.1	6	7.5	0
Total	2360					245.6

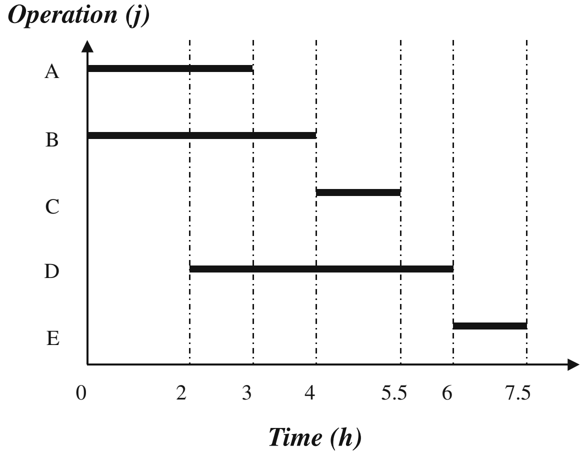


Fig. 4.8 Ganntt chart for the case study (Majozi, 2005)

contaminant, since they all produce sodium chloride (NaCl) as a byproduct. This byproduct is removed from the organic phase using liquid–liquid extraction in which freshwater is introduced to form an aqueous phase. However, in the B and D operations water is also used as a solvent. In the C and E operations, water is used for polishing rather than extraction purposes, hence the zero contaminant loads. Each of the five operations belongs to a multi-stage batch process. Prior to the exploration of water reuse/recycle opportunities, these operations used 2360 kg of freshwater in a 7.5 h time horizon as shown in Table 4.2. The objective function in all the following four scenarios is the minimization of freshwater requirement.

Water Reuse/Recycle Module

Since the formulation of the constraints has been presented in detail in Section 4.3 using, only the new constraints will be presented in this section. The new constraints were necessitated by the existence of operations C and E for which the contaminant mass load is zero as aforementioned. Without any modifications in the presented mathematical formulations for scenarios 1 and 3, i.e. fixed outlet concentration, this condition would suggest that there is no need for the utilization of water in operations C and E (see constraints (4.3)). However, water is required in these operations for polishing purposes, although this is not associated with any contaminant removal. The minimum amount of water required in these operations is 300 kg. Therefore, the following new constraints is added to the mathematical formulations for scenarios 1 and 3.

$$m_u\left(s_{\text{in},j},p\right) \geq 300y\left(s_{\text{in},j},p\right), \forall p \in P, j \in J, j = C,E \tag{4.52}$$

In scenarios 2 and 4, i.e. fixed water quantity and existence of reusable water storage, the following constraints is necessary to ensure that the capacity of reusable water storage is taken into account. The reusable water storage cannot contain more than 800 kg.

$$q_s(p) \leq 800, \forall p \in P \tag{4.53}$$

Sequencing/Scheduling Module

Sequencing of operations was performed over a 7.5 h time horizon, i.e. $H = 7.5$. Four time points were used in scenarios 1 and 2, and seven time points proved optimal for scenarios 3 and 4. All the constraints are as presented in detail in Section 4.5.

Computational Results

Scenario 1: Formulation for fixed outlet concentration without reusable water storage

The overall model for scenario 1, which is MILP, entails 1320 constraints, 546 continuous and 120 discrete/binary variables. 52 nodes were explored in the branch and bound algorithm and the optimal freshwater requirement of 1767.84 kg was reached in 1.61 CPU seconds. Figure 4.9 shows the corresponding water reuse/recycle network.

Figure 4.9 shows that 1767.84 kg of freshwater is required over the 7.5 h time horizon. This corresponds to 25% reduction in freshwater requirement compared to the situation without water recycle/reuse. Although water from process A is at a relatively lower concentration of 0.1 kg salt/kg water, the time constraints in the absence

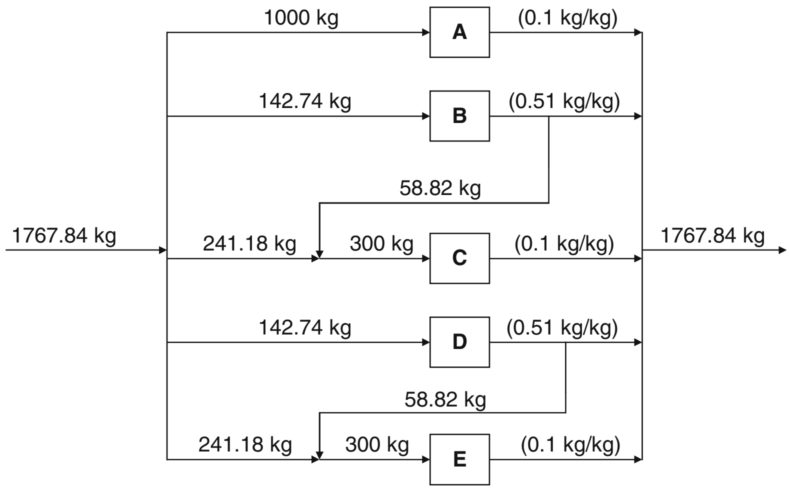


Fig. 4.9 Water reuse/recycle network for scenario 1 – first case study (Majozi, 2005)

of reusable water storage forbids any possibility for recycle/reuse. According to the Gantt chart in Fig. 4.8, none of the other operations commences when operation A finishes at 3 h. Consequently, all the wastewater from operation A is dispensed with as effluent. However, the time constraints allows water reuse from operations B and D to operations C and E, respectively. The concentration constraints is met by diluting the water reuse streams with freshwater to reduce concentration from 0.51 to 0.1 kg salt/kg water.

Scenario 2: Formulation for fixed water quantity without reusable water storage

The formulation for this scenario entails 1411 constraints, 511 continuous and 120 binary variables. The reduction in continuous variables compared to scenario 1 is due to the absence of linearization variables, since no attempt was made to linearize the scenario 2 model as explained in Section 4.3. An average of 1100 nodes were explored in the branch and bound search tree during the three major iterations between the MILP master problem and the NLP subproblem. The problem was solved in 6.54 CPU seconds resulting in an optimal objective of 2052.31 kg, which corresponds to 13% reduction in freshwater requirement. The corresponding water recycle/reuse network is shown in Fig. 4.10.

Scenario 3: Formulation for fixed outlet concentration with reusable water storage

The corresponding mathematical formulation entails 5534 constraints, 1217 continuous and 280 binary variables. An average of 4000 nodes were explored in the branch and bound search tree. The solution required three major iterations and took 309.41 CPU seconds to obtain the optimal solution of 1285.50 kg. This corresponds to 45.53% reduction in freshwater demand. A water reuse/recycle network that corresponds to this solution is shown in Fig. 4.11.

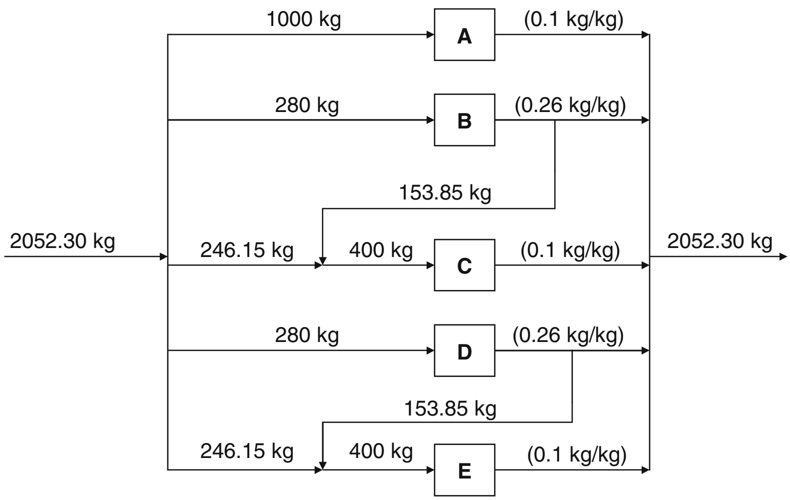


Fig. 4.10 Water reuse/recycle network for scenario 2 – first case study (Majozi, 2005)

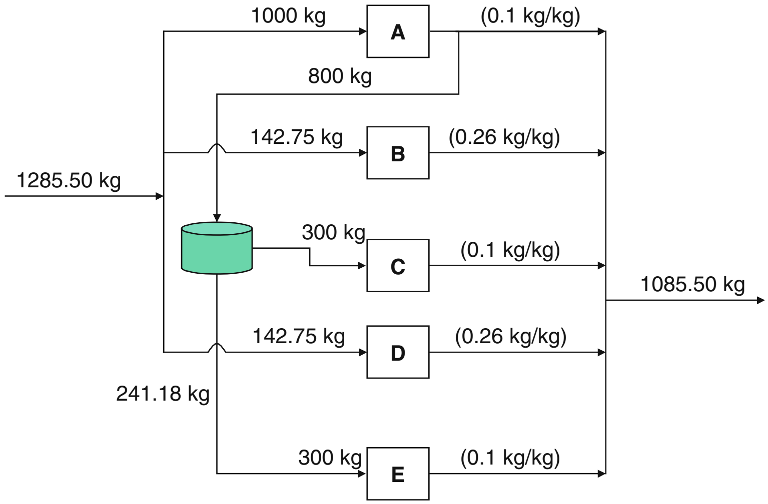


Fig. 4.11 Water reuse/recycle network for scenario 3 – first case study (Majozi, 2005)

It is evident from Fig. 4.11 that the existence of central reusable water storage allows the time constraints to be overridden, thereby providing an opportunity for reuse from operation A to operations C and E. This is the main reason for the significant reduction in freshwater requirement. Scenario 1, which is similar to scenario 3 without the presence of reusable water storage, resulted in 25% instead of 45.53% freshwater reduction. It is worth noting that the solution suggests a smaller

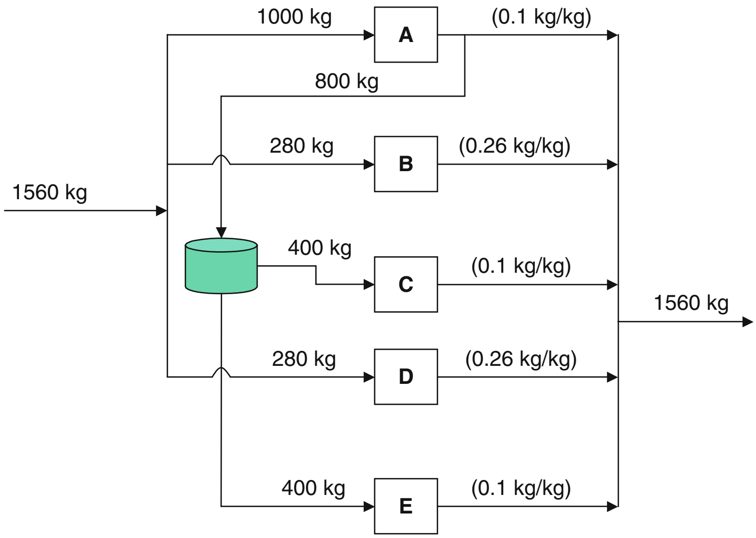


Fig. 4.12 Water reuse/recycle network for scenario 4 – first case study

wastewater than freshwater stream, viz. 1085.50 kg vs 1285.50 kg. This is due to the fact that some reusable water remains in reusable water storage tank.

Scenario 4: Formulation for fixed water quantity with reusable water storage

The overall model for this scenario involves 5614 constraints, 1132 continuous 280 binary variables. Three major iterations with an average of 1200 nodes in the branch and bound search tree were required in the solution. The objective value of 1560 kg, which corresponds to 33.89% reduction in freshwater requirement, was obtained in 60.24 CPU seconds. An equivalent of this scenario, without reusable water storage, i.e. scenario 2, resulted in 13% reduction in fresh water. Figure 4.12 shows the water recycle/reuse network corresponding to this solution.

4.7 Second Case Study

The second case study presents a situation in which the duration time, instead of starting and finishing times, is provided in the problem specification. This is typical of practical scheduling environment where the starting and finishing times are dependent on a broader scheduling framework. This case study involves two single-stage batch processes wherein each stage is a liquid–liquid extraction process using fresh water for the removal of a salt byproduct (contaminant) from the organic phase. The organic phase is brought into the facility as a raw material at a reasonable cost. The wastewater streams from the extraction processes are collected into a common water treatment facility before disposal. Table 4.3 shows all the necessary data for the mathematical formulation, including cost information. Figure 4.13 shows the indices allocated to freshwater and raw streams in the mathematical formulation. In this case study, the objective is to maximise operating

Table 4.3 Data for the second case study

Operation (j)	Capacity (t)	Suitability	Duration (h)	$C_{out}^{U}(j)$ (kg/t)	$C_{in}^{U}(j)$ (kg/t)	$M(j)$ (kg)
1	200	Extraction	2	0.1	0.05	2
2	200	Extraction	1	0.2	0.1	5
Cost data						
Cost ($/t)	Product		Raw		Wastewater	
	$s2_{out,1}$	$s2_{out,2}$	$s2_{in,1}$	$s2_{in,2}$	$s1_{out,1}$	$s1_{out,2}$
Selling price	2300	2000	–	–	–	–
Cost	–	–	108	82	–	–
Cost	–	–			500	500

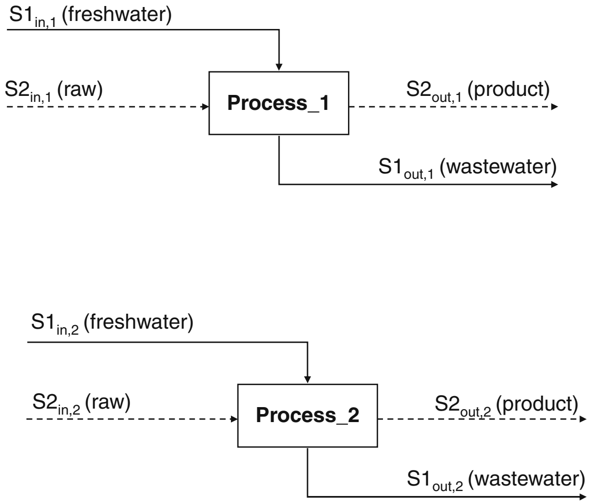


Fig. 4.13 Indices used for the streams in the second case study

profit by minimizing the operating costs whilst maximizing revenue over a 6 h time horizon and is stated as follows.

Maximise

$$\sum_{p}\left(\sum_{\substack{s2_{\text{out},j}\\ j=1,2}} SP(j)m_p\left(S_{\text{out}},j,p\right)-\sum_{\substack{s2_{\text{in},j}\\ j=1,2}} CR(j)\Psi(j)m_u\left(S_{\text{in, jp}}\right)-CE\sum_{s1_{\text{out},j}} m_p\left(S_{\text{out, j,p}}\right)\right) \tag{4.54}$$

The operating costs entail raw material costs and effluent treatment costs. Only scenarios 1 and 3, i.e. fixed outlet concentration with and without reusable water storage, are considered. The additional information provided for the case study pertains to the mass ratios between raw material streams and freshwater. In process 1, 1 kg of water (aqueous phase) is required to wash 3 kg of raw material stream (organic phase) to the desired specification. In process 2, 1 kg of water is required for every 2 kg of raw material stream. These requirements are dictated by mass transfer.

Scenario 1: Fixed outlet concentration without reusable water storage

Since the capacity of operating units and mass ratios between freshwater and raw streams are provided, the following additional constraints are necessary.

Capacity Constraints

These constraints ensure that the overall input into each of the units does not exceed the capacity of the units, i.e. 200 t.

$$4m_u\left(s1_{\text{in},1},p\right) \le 200y\left(s1_{\text{in},1},p\right), \forall p \in P \tag{4.55}$$

$$3m_u\left(s1_{\text{in},2},p\right) \le 200y\left(s1_{\text{in},2},p\right), \forall p \in P \tag{4.56}$$

Mass Ratio Constraints

These constraints ensure that the mass ratio between freshwater and raw material is obeyed so as to fulfil the mass transfer requirements. Using simple algebra, the given ratios can readily be stated as follows.

$$3m_p\left(s1_{\text{out},1},p\right) = m_p\left(s2_{\text{out},1},p\right) + 0.002y\left(s1_{\text{in},1},p-1\right), \forall p \in P, p > p_1 \tag{4.57}$$

$$2m_p\left(s1_{\text{out},2},p\right) = m_p\left(s2_{\text{out},2},p\right) + 0.005y\left(s1_{\text{in},2},p-1\right), \forall p \in P, p > p_1 \tag{4.58}$$

Computational Results

Table 4.4 is the summary of the mathematical model and the results obtained for the case study. The model for scenario 1 involves 637 constraints, 245 continuous and 42 binary variables. Seventy nodes were explored in the branch and bound algorithm. The model was solved in 1.61 CPU seconds, yielding an objective value (profit) of $1.61 million over the time horizon of interest, i.e. 6 h. This objective is concomitant with the production of 850 t of product and utilization of 210 t of freshwater. Ignoring any possibility for water reuse/recycle, whilst targeting the same product quantity would result in 390 t of freshwater utilization. Therefore, exploitation of water reuse/recycle opportunities results in more than 46% savings in freshwater utilization, in the absence of central reusable water storage. The water network to achieve the target is shown in Fig. 4.14.

Table 4.4 Second case study results

	Scenario 1	Scenario 2
Problem structure	MILP	MINLP
Number of constraints	637	1195
Continuous variables	245	352
Discrete variables	42	70
Nodes in B&B	70	151
Major iterations	–	3
Objective value ($M)	1.61	1.67
CPU time (s)	0.57	2.48
Product (tons)	849.96	849.96
Freshwater (tons)	210	185
Percentage of savings in freshwater (relative to no reuse/recycle option)	46.15	52.56

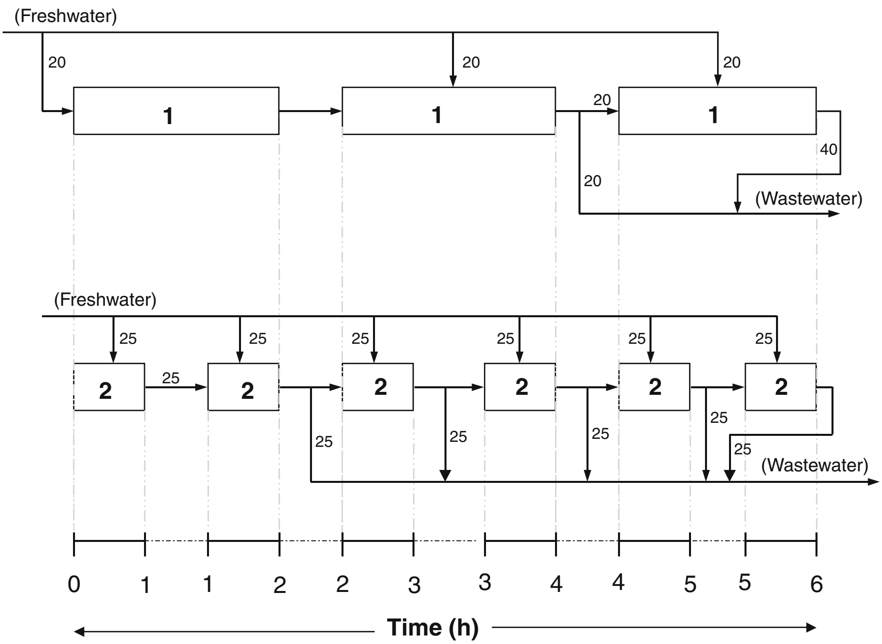


Fig. 4.14 Water recycle/reuse network for the second case study – *scenario 1* (Majozi, 2005)

Figure 4.14 stipulates that three batches of process 1 and six batches of process 2 have to be completed over the chosen time horizon. Only recycle rather than reuse opportunities are exploited in both processes, since each batch either utilises freshwater or recycled water from the preceding batch of the same process. The quantities of recycled water are shown in the diagram.

As shown in Table 4.4, the model for scenario 2, which is a nonconvex MINLP, consists of 1195 constraints, 352 continuous and 70 binary variables. An average of 151 nodes were explored in the branch and bound algorithm over the 3 major iterations between the MILP master problem and NLP subproblems. The problem was solved in 2.48 CPU seconds with an objective value of \$1.67 million. Whilst the product quantity is the same as in scenario 1, i.e. 850 t, the water requirement is only 185 t, which corresponds to 52.56% reduction in freshwater requirement. The water network to achieve this target is shown in Fig. 4.15.

Figure 4.15 shows the exploitation of water reuse and recycle opportunities to achieve the target of 185 t of freshwater. Water from the second and fourth batches of process 2 is reused in the second and third batches of process 1, respectively. Water from storage is reused in the fourth and fifth batches of process 2. The rest of the batches either utilize freshwater or recycled water from preceding batches of the same process.

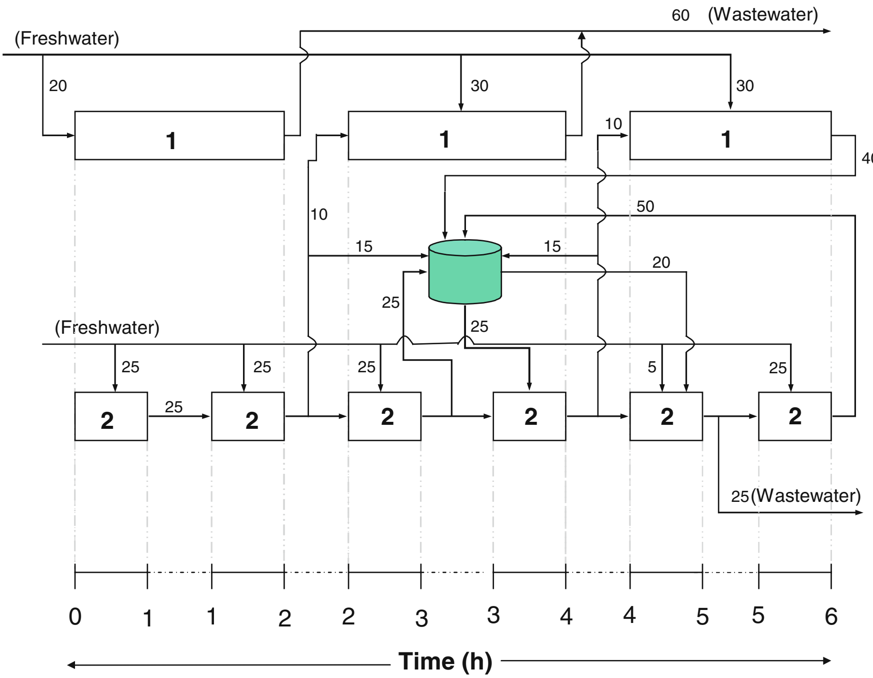


Fig. 4.15 Water recycle/reuse network for the second case study – *scenario 3* (Majozi, 2005)

4.8 Concluding Remarks

A mathematical formulation based on uneven discretization of the time horizon for the reduction of freshwater utilization and wastewater production in batch processes has been developed. The formulation, which is founded on the exploitation of water reuse and recycle opportunities within one or more processes with a common single contaminant, is applicable to both multipurpose and multiproduct batch facilities. The main advantages of the formulation are its ability to capture the essence of time with relative exactness, adaptability to various performance indices (objective functions) and its structure that renders it solvable within a reasonable CPU time. Capturing the essence of time sets this formulation apart from most published methods in the field of batch process integration. The latter are based on the assumption that scheduling of the entire process is known a priori, thereby specifying the start and/or end times for the operations of interest. This assumption is not necessary in the model presented in this chapter, since water reuse/recycle opportunities can be explored within a broader scheduling framework. In this instance, only duration rather start/end time is necessary. Moreover, the removal of this assumption allows problem analysis to be performed over an unlimited time horizon. The specification of start and end times invariably sets limitations on the time horizon over which water reuse/recycle opportunities can be explored. In the four scenarios explored in this chapter, the model was solved within 3 CPU seconds in a 1.82 GHz, Pentium 4 processor. Applications of the model to a published literature example and two practical case studies have shown that the existence of storage greatly improves water minimisation opportunities, with savings in freshwater demand in excess of 50%. It is worthy of mention, however, that the presented methodology is only applicable to single contaminant media. The issue of multiple contaminants and multipurpose batch plants is addressed in Chapter 6 of this textbook.

4.9 Exercise

Task: Revisit the first case study and minimise wastewater for the 4 presented scenarios, where only duration for tasks is known, i.e. the start and ending times are variables rather than parameters. Consider time horizon to be fixed at 7.5 h. Does this improve or deteriorate the results? Explain.

References

Glover, F., 1975. Improved linear programming formulations of nonlinear integer problems. Manag. Sci., 22(4): 455–460.

Majozi, T., 2005. Wastewater minimization using central reusable water storage in batch processes. Comput. Chem. Eng., 29(7): 1631–1646.

Majozi, T., Brouckaert, C.J.B., Buckley, C.A.B., 2006. A graphical technique for wastewater minimisation in batch processes. J. Environ. Manag., 78(4): 317–329.

Majozi, T., Zhu, X.X., 2001. A novel continuous time MILP formulation for multipurpose batch plants. 1. Short-term scheduling. Ind. Eng. Chem. Res., 40(25): 5935–5949.

Wang, Y.P., Smith, R., 1995. Time pinch analysis. Trans. IChemE, 73a: 905–914.

Chapter 5
Storage Design for Maximum Wastewater Reuse in Batch Plants

Overview In Chapter 4, an approach for wastewater minimisation through exploration of recycle and reuse opportunities was presented with the implicit assumption that the capacity of reusable water storage was fixed. This chapter presents an algorithm for minimisation of reusable water storage in batch plants (Majozi, Ind. Eng. Chem. Res., 45(17): 5936–5943, 2006). This algorithm is applicable to both multipurpose and multiproduct batch plants and is based on a two-stage technique, with each stage focussing on a dedicated objective function. In the first stage, the objective is the minimisation of freshwater, given the maximum reusable water storage capacity. Once the minimum freshwater target has been set, it is then fixed and used as an input parameter in the second stage. In the second stage, central reusable water storage capacity is a variable that needs to be minimised subject to the minimum freshwater target set in the first stage. The overall formulation is based on uneven discretization of the time horizon approach developed by Majozi and Zhu (2001) using a so-called state sequence network, and is an extension of the material presented in Chapter 4 of this textbook. In this chapter, the algorithm is applied to a case study in which it yields more than 45% savings in freshwater demand compared to the case without the exploitation of water recycle and reuse. Moreover, more than 60% reduction in reusable water storage capacity is observed in comparison to the situation in which central reusable water storage capacity is fixed beforehand. Although only a single contaminant case is addressed in this chapter, the presented methodology is also applicable to a multiple contaminant environment. Worthy of mention is that the structure of the resultant mathematical formulation is such that global optimality cannot be guaranteed.

5.1 The Essence of the Problem

It is well known that storage plays a significant role in bypassing the time dimension that is inherent in batch facilities. This allows for reuse and recycle of water across different time intervals within a given time horizon, thereby reducing the overall freshwater demand and wastewater generation. This is depicted in Fig. 5.1,

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_5,

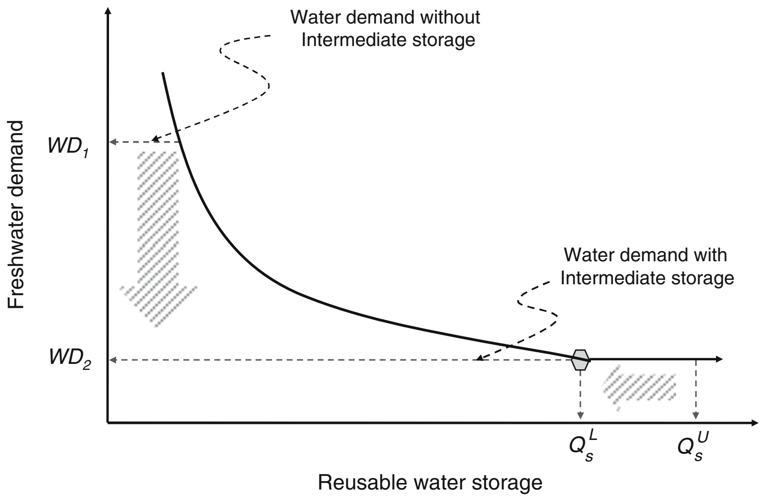


Fig. 5.1 Impact of reusable water storage on freshwater demand (Majozi, 2006)

where WD_1 is the freshwater demand in the absence of intermediate water storage and WD_2 is the freshwater demand in the presence of storage. It is worthy of mention, however, that absence of intermediate storage does not necessarily forbid the exploitation of direct recycle and reuse opportunities if time and concentration constraints allow. For example, a process that is finishing at time t can always directly supply reusable water to another process starting at time t, as long as concentration constraints are not violated. The presence of intermediate storage further improves opportunities for water recycle and reuse, since water that cannot be directly reused or recycled can be stored for later use.

In situations characterized by significant physical constraints, which is a common encounter in batch production facilities, the size of reusable water storage might surface as one of the significant causes for concern. In these situations, the minimum storage capacity, which might involve total elimination, is mandatory. Figure 5.1 shows that there exists the minimum amount of storage, Q_s^L, beyond which the overall water demand for the operations involved is not affected. In most wastewater minimisation formulations, the upper bound in water storage, i.e. Q_s^U, which is usually known a priori, is used during the optimisation of freshwater use and wastewater generation. However, this might be much larger than the minimum water storage that corresponds to the minimum freshwater use and wastewater generation. This particular observation has always been overlooked in literature. It is worthy of note in Fig. 5.1 that the objectives which are being addressed are of a conflicting nature, since minimisation of freshwater use requires increase in reusable water storage.

5.2 Problem Statement

More succinctly, the problem addressed in this chapter can be stated as follows. For each water using operation, given:

(i) the contaminant mass load,
(ii) the water requirement,
(iii) the duration or start and finish times to achieve the desired effect, e.g. mass transfer, degree of cleanliness of the vessel, etc.,
(iv) maximum *potential* reusable water storage,
(v) maximum inlet and outlet water concentrations and
(vi) time horizon of interest,

determine the minimum reusable water storage which is concomitant with *minimum* freshwater requirement or wastewater generation. In this instance, the minimum reusable water storage could correspond to the complete elimination of reusable water storage, provided that the minimum freshwater requirement is not compromised. It is mainly for this reason that reusable water storage is referred to as potential rather than existing in condition (iv). In condition (iii), if duration instead of start and finish times is given, then the minimization of storage should be considered within an overall scheduling framework in which the start and finish times become optimisation variables (Majozi, 2005, 2006).

5.3 Mathematical Model

The mathematical model for the problem addressed in this chapter entails the following sets, variables and parameters.

Sets

S	$\{s \mid s \text{ is a state}\} = S_{\text{in},j} \cup S_{\text{out},j}$
J	$\{j \mid j \text{ is a unit}\}$
P	$\{p \mid p \text{ is a time point}\}$
$S_{\text{in},j}$	$\{s_{\text{in},j} \mid s_{\text{in},j} \text{ is an input state to unit j}\}$
$S_{\text{out},j}$	$\{s_{\text{out},j} \mid s_{\text{out},j} \text{ is an output state from unit j}\}$

Variables

$C_{out}\,(j,p)$	outlet concentration from unit j at time point p
$C_{in}\,(j,p)$	inlet concentration to unit j at time point p
$CS_{out}\,(p)$	outlet concentration from storage at time point p
$CS_{in}\,(p)$	inlet concentration to storage at time point p
$d\,(s,p)$	amount of state s delivered to customers at time point p, $s \in S_{out,j}$
$m_e\,(s,p)$	amount of state s dispensed with as effluent at time point p, $s \in S_{out,j}$
$m_f\,(s,p)$	amount of fresh water used in unit j at time point p, $s \in S_{in,j}$
$m_p\,(s,p)$	amount of state s produced at time point p, $s \in S_{out,j}$
$m_u\,(s,p)$	amount of state s used at time point p, $s \in S_{in,j}$
$m_r\,(j,j',p)$	amount of water recycled or reused between two units j and j' at time point p
$ms_{in}\,(s,p)$	amount of state s that is transferred to storage at time point p, $s \in S_{out,j}$
$ms_{out}\,(s,p)$	amount of state s that is transferred from storage to a particular unit j at time point p, $s \in S_{in,j}$
$q_s\,(p)$	amount of water stored at time point p
$t_p\,(s,p)$	time at which state s is produced at time point p, $s \in S_{out,j}$
$t_r\,(j,j',p)$	time at which water is recycled or reused between two units j and j' at time point p
$t_u\,(s,p)$	time at which state s is used at time point p, $s \in S_{in,j}$
$ts_{in}\,(s,p)$	time at which state s is transferred to storage from operation j at time point p, $s \in S_{out,j}$
$ts_{out}\,(s,p)$	time at which state s is transferred to operation j from storage at time point p, $s \in S_{in,j}$
$y\,(s,p)$	binary variable associated with usage of state s at time point p, $s \in S_{in,j}$
$y\,(j,j',p)$	binary variable associated with recycle or reuse between two units j and j' at time point p
$ys_{in}\,(s,p)$	binary variable associated with the transfer of state s from operation j to storage at time point p, $s \in S_{out,j}$
$ys_{out}\,(s,p)$	binary variable associated with the transfer of state s from storage to operation j at time point p, $s \in S_{in,j}$

Parameters

$C^U_{out}\,(j)$	maximum outlet contaminant concentration from unit j
$C^U_{in}\,(j)$	maximum inlet contaminant concentration to unit j
H	time horizon of interest
$M\,(j)$	mass-load of contaminant in unit j

$W^U(j)$	maximum water requirement in unit j
$W^L(j)$	minimum water requirement in unit j
$Q_s^0(s)$	initial amount of state $s$stored
Q_s^U	maximum capacity of reusable water storage
Q_s^L	minimum capacity of reusable water storage
V_j	capacity of a particular unit j
$W(j)$	fixed water requirement in unit j

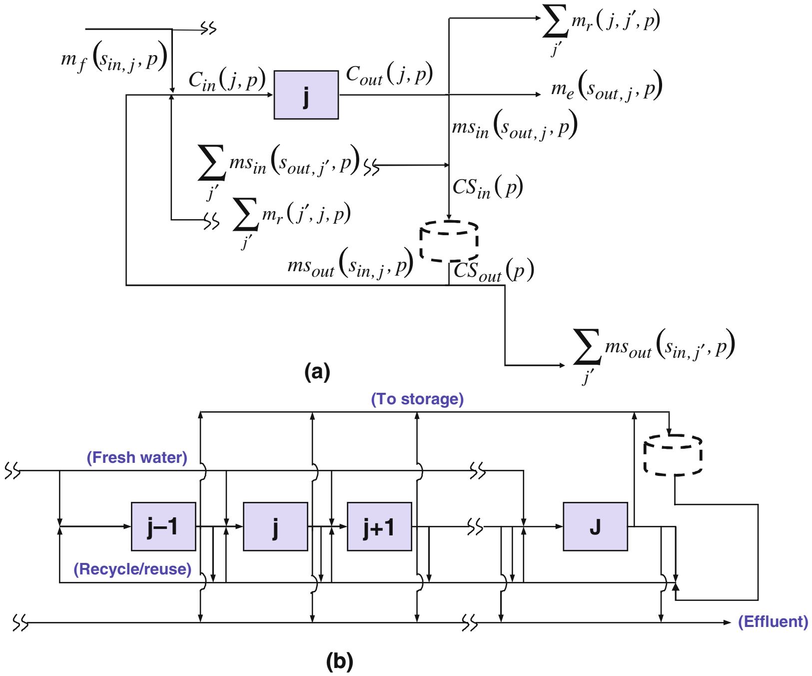


Fig. 5.2 Superstructure for the mathematical formulation with *potential* reusable water storage (Majozi, 2006)

The mathematical model presented in this section is based on the superstructure given in Fig. 5.2 for wastewater minimisation using central reusable water storage of fixed capacity. The superstructure represents a situation where there is potential for reusable water storage. In this situation, water used in each water using operation j can be supplied from the fresh water header, the recycle/reuse water header, the reusable water storage header or a combination of the three headers. Water from each operation j can be recycled to the same operation, reused in downstream processes, transferred to reusable water storage and/or removed as effluent. If minimum freshwater requirement can be attained without reusable water storage, the option of receiving water from or transferring water to storage is nullified. Only water streams are shown for clarity purposes. The index p that appears in all the variables shown in the superstructure captures the essence of time. Figures 4.2a, b show an exploded and condensed view of the superstructure, respectively. Each of the water using operations shown in the superstructure could either exist as an isolated entity or form part of a complete batch chemical process. Different water using operations in the superstructure could belong to the same or distinct processes. The other process units in a complete batch plant are deliberately omitted from the diagram, since the focus of the mathematical formulation is only on water operations.

This mathematical model is made up of two sets of constraints that are built within the same framework. One set of constraints focuses on the exploration of water reuse/recycle opportunities and the other on proper sequencing to capture the time dimension. Although this model has been presented in detail in Chapter 4, it is presented here in sufficient detail to facilitate understanding.

5.3.1 Water Reuse/Recycle Constraints

In exploring the recycle and reuse opportunities within a complete batch process, two mass-transfer scenarios are mathematically formulated in the following sections. The first scenario is based on fixed outlet concentration and fixed contaminant mass load from each water using operation. In this scenario, the outlet concentration is always the maximum possible in a given water using operation. This situation allows for the quantity of water used in the operation to vary from the limiting water requirement. The limiting water requirement is the amount of water required if the initial contaminant concentration in a water using operation corresponds to the maximum permissible concentration. The second scenario is based on fixed contaminant mass load and fixed water requirement for each water using operation. In this situation, the inlet and outlet water concentrations are allowed to vary within predefined bounds. More succinctly, in the first scenario outlet concentration is a parameter and flowrate a variable, whilst in the second scenario the opposite is true. However, in both scenarios, the mass load is fixed. The practical relevance of these scenarios can be justified by noting the following two examples.

The first scenario is typical of a washing operation, wherein the contaminant amount is fixed, e.g. washing a process vessel that is used in a fixed recipe. In this case, using less amount of water will result in higher concentration of contaminant in the wastewater stream for a fixed contaminant mass load. However, there is a limit to this outlet concentration, which is set by process conditions like saturation point, corrosion issues, etc. This limit determines the minimum amount of freshwater that can be used. The second scenario, on the other hand, is typical of a liquid–liquid extraction operation, wherein for a given amount of organic phase a certain amount of aqueous phase is required to remove a fixed amount of contaminant. This scenario is a common encounter in agrochemical operations where organic solvents are used as reaction media and water used to remove inorganic salts that form as byproducts during the reactions. In this situation, the amounts of the aqueous (water) and organic phases are determined by the capacity of the equipment used. It is evident that in this scenario the outlet concentration need not be fixed at any level, but should not exceed the maximum allowed.

Scenario 1: *Formulation for fixed outlet concentration and fixed contaminant mass load*

This formulation is based on a superstructure given in Fig. 5.2.

$$m_u\left(s_{\text{in},j},p\right) = \sum_{j'\in J} m_r\left(j',j,p\right) + m_f\left(s_{\text{in},j},p\right), \quad \forall p \in P, s_{\text{in},j} \in S_{\text{in},j} \tag{5.1}$$

$$m_p\left(s_{\text{out},j},p\right) = m_e\left(s_{\text{out},j},p\right) + \sum_{j'\in J} m_r\left(j,j',p\right), \quad \forall p \in P, s_{\text{out},j} \in S_{\text{out},j} \tag{5.2}$$

$$\begin{aligned} m_p\left(s_{\text{out},j},p\right) C_{\text{out}}(j,p) = {} & m_u\left(s_{\text{in},j},p-1\right) C_{\text{in}}(j,p-1) \\ & + M(j)y\left(s_{\text{in},j},p-1\right), \\ & \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j}, p \in P, p > p_1 \end{aligned} \tag{5.3}$$

$$C_{\text{in}}(j,p) = \frac{\sum_{j'} m_r\left(j',j,p\right) C_{\text{out}}(j',p)}{\sum_{j'} m_r\left(j',j,p\right) + m_f\left(s_{\text{in},j},p\right)}, \forall j,j' \in J, p \in P, s_{\text{in},j} \in S_{\text{in},j} \tag{5.4}$$

$$C_{\text{out}}(j,p) = C_{\text{out}}^{U}(j) y\left(s_{\text{in},j},p-1\right), \forall j \in J, p \in P, p > p_1, s_{\text{in},j} \in S_{\text{in},j} \tag{5.5}$$

$$W^L y\left(s_{\text{in},j},p\right) \le m_u\left(s_{\text{in},j},p\right) \le W^U(j)\, y\left(s_{\text{in},j},p\right), \forall s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{5.6}$$

Constraints (5.1) states that the inlet stream into any operation j is made up of recycle/reuse stream, fresh water stream and a stream from reusable water storage. On the other hand, the outlet stream from operation j can be removed as effluent, reused in other processes, recycled to the same operation and/or sent to reusable water storage as shown in constraints (5.2). Constraints (5.3) is the mass balance around unit j. It states that the contaminant mass-load difference between outlet and inlet streams for the same unit j is the contaminant mass-load picked up in unit j. The inlet concentration into operation j is the ratio of the contaminant amount in the inlet stream and the quantity of the inlet stream as stated in constraints (5.4). The amount of contaminant in the inlet stream to operation j consists of the contaminant in the recycle/reuse stream and the contaminant in the reusable water storage stream. Constraints (5.5) states that the outlet concentration from any unit j is fixed at a maximum predefined concentration corresponding to the same unit. It should be noted that streams are expressed in quantities instead of flowrates, which is indicative of any batch operation. The total quantity of water used at any point in time must be within bounds of the equipment unit involved as stated in constraints (5.6). Following are the storage-specific constraints.

$$qs\,(p) = qs\,(p-1) + \sum_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j}, p\right) - \sum_{s_{\text{in},j}} ms_{\text{out}}\left(s_{\text{in},j}, p\right),$$
$$\forall j \in J, p \in P, p > p_1, s_{\text{out},j} \in S_{\text{out},j}, s_{\text{in},j} \in S_{\text{in},j} \tag{5.7}$$

$$qs(p_1) = Q_s^0 - \sum_{s_{\text{in},j}} ms_{\text{out}}(s_{\text{in},j}, p_1),\ \forall j \in J, s_{\text{in},j} \in S_{\text{in},j} \tag{5.8}$$

$$qs\,(p) \le Q_s^U,\ \forall p \in P \tag{5.9}$$

$$CS_{\text{in}}\,(p) = \frac{\sum_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j}\right) C_{\text{out}}\,(j,p)}{\sum_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j}, p\right)},$$
$$\forall j \in J, p \in P, s_{\text{out},j} \in S_{\text{out},j} \tag{5.10}$$

$$CS_{\text{out}}\,(p) = \frac{qs\,(p-1)\,CS_{\text{out}}\,(p-1) + \sum_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j}\right) C_{\text{out}}\,(j,p)}{qs\,(p-1) + \sum_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j}, p\right)},$$
$$\forall j \in J, p \in P, p > p_1, s_{\text{out},j} \in S_{\text{out},j} \tag{5.11}$$

$$CS_{\text{out}}\,(p_1) = CS_{\text{out}}^0 \tag{5.12}$$

Constraints (5.7) is the mass balance around reusable water storage tank. It states that the amount stored at any time point p is determined by the amount stored at the previous time point $p-1$ plus the difference between the quantity transferred from and the quantity transferred to the water using operations at time point p. However, at the beginning of the time horizon of interest, none of the water using operations is complete and ready to transfer to storage. Also, the amount stored at the previous time point corresponds to initial amount of reusable water available in storage Q_s^0. Therefore, at the beginning of the time horizon, constraints (5.8) replaces constraints (5.7). Constraints (5.9) ensures that the amount of water stored at any point in time does not exceed the capacity of reusable water storage.

Constraints (5.10) and (5.11) respectively give the inlet and outlet concentrations for the reusable water storage tank. At any given time point p, the inlet concentration is the ratio of the contaminant mass load in all streams transferred from water using operations to the overall quantity of the stream transferred to reusable water storage tank. The outlet concentration at any time point p is defined as the contaminant load at the previous time point $p-1$ plus the contaminant load in the incoming stream from water using operations divided by the total quantity of reusable water in the storage tank. This is also the definition of the contaminant concentration inside the reusable water storage tank. Hence, it is assumed that the outlet concentration from the storage tank is the same as the concentration inside the tank. This is indeed a valid assumption if perfect mixing is achieved within the tank. It is evident that constraints (5.11) is not applicable at the beginning of the time horizon of interest for reasons similar to constraints (5.7). Therefore, constraints (5.12) replaces constraints (5.11) at the beginning of the time horizon. These constraints constitute a complete water reuse/recycle mathematical model, which is a nonconvex MINLP due to the bilinear terms encountered in constraints (5.3), (5.4), (5.10) and (5.11).

Scenario 2: *Formulation for fixed water quantity and fixed contaminant mass load*

In a situation where the quantity of water required in any given operation is fixed, constraints (5.5) and (5.6) have to be modified as follows.

$$C_{\mathrm{out}}(j,p) \le C_{\mathrm{out}}^{U}(j)\, y\left(s_{\mathrm{in},j}, p-1\right), \quad \forall j \in J, p \in P, p > p_1, s_{\mathrm{in},j} \in S_{\mathrm{in},j} \tag{5.13}$$

$$m_u\left(s_{\mathrm{in},j}, p\right) = W^{U}(j)\, y\left(s_{\mathrm{in},j}, p\right), \forall s_{\mathrm{in},j} \in S_{\mathrm{in},j}, p \in P \tag{5.14}$$

The foregoing constraints only consider material balances. However, batch operations are characterized by time-dependent activities which require dedicated constraints. These are given in the following section.

5.3.2 Sequencing/Scheduling Constraints

The sequencing set of constraints focuses on capturing the time dimension, which is intrinsic in batch operations. The following constraints, which apply irrespective of the chosen scenario (scenario 1 or scenario 2), constitute the scheduling set of constraints for the proposed mathematical model.

$$y_r\left(j,j',p\right) \le y\left(s_{\mathrm{in},j'}, p\right), \quad \forall s_{\mathrm{in},j'} \in S_{\mathrm{in},j'}, p \in P, j,j' \in J \tag{5.15}$$

$$t_r\left(j,j',p\right) \le t_p\left(s_{\mathrm{out},j}, p\right) + H\left(1 - y_r\left(j,j',p\right)\right), \quad \forall j,j' \in J, s_{\mathrm{out},j} \in S_{\mathrm{out},j}, p \in P \tag{5.16}$$

$$t_r\left(j,j',p\right) \ge t_p\left(s_{\mathrm{out},j}, p\right) - H\left(1 - y_r\left(j,j',p\right)\right), \quad \forall j,j' \in J, s_{\mathrm{out},j} \in S_{\mathrm{out},j}, p \in P \tag{5.17}$$

$$t_r\left(j,j',p\right) \le t_u\left(s_{\mathrm{in},j'}, p\right) + H\left(1 - y_r\left(j,j',p\right)\right), \quad \forall j,j' \in J, s_{\mathrm{in},j'} \in S_{\mathrm{in},j'}, p \in P \tag{5.18}$$

$$t_r\left(j,j',p\right) \ge t_u\left(s_{\mathrm{in},j'}, p\right) - H\left(1 - y_r\left(j,j',p\right)\right), \quad \forall j, j' \in J, s_{\mathrm{in},j'} \in S_{\mathrm{in},j'}, p \in P \tag{5.19}$$

$$t_u\left(s_{\mathrm{in},j}, p\right) \ge t_p\left(s_{\mathrm{out},j}, p'\right) - H\left(2 - y\left(s_{\mathrm{in},j}, p\right) - y\left(s_{\mathrm{in},j}, p'-1\right)\right), \quad \forall j \in J, s_{\mathrm{in},j} \in S_{\mathrm{in},j}, s_{\mathrm{out},j} \in S_{\mathrm{out},j}, \forall p,p' \in P, p' > p_1, p \ge p' \tag{5.20}$$

$$t_u\left(s_{\mathrm{in},j}, p\right) \ge t_p\left(s_{\mathrm{in},j}, p'\right) - H\left(2 - y\left(s_{\mathrm{in},j}, p\right) - y\left(s_{\mathrm{in},j}, p'\right)\right), \quad \forall j \in J, s_{\mathrm{in},j} \in S_{\mathrm{in},j}, p,p' \in P, p \ge p' \tag{5.21}$$

$$t_u\left(s_{\mathrm{out},j}, p\right) \ge t_p\left(s_{\mathrm{out},j}, p'\right) - H\left(2 - y\left(s_{\mathrm{out},j}, p\right) - y\left(s_{\mathrm{out},j}, p'\right)\right), \quad \forall j \in J, s_{\mathrm{out},j} \in S_{\mathrm{out},j}, p,p' \in P, p \ge p' \tag{5.22}$$

Constraints (5.15) states that if water is reused from operation j to operation j' at a given time point p, then operation j' should commence at time point p. However, the fact that operation j' commences at time point p does not necessarily mean that there is a corresponding recycle/reuse stream at time point p. This is due to the fact that operation j' could be using freshwater or water from storage instead of recycle/reuse stream from another process j. Constraints (5.16) and (5.17) together ensure that water recycle/reuse from operation j to operation j' coincides with the completion of operation j at time point p. Similarly, constraints (5.18) and (5.19) ensure that water recycle/reuse from operation j to operation j' coincides with the start of operation j' at time point p. Constraints (5.20) states that any operation j will start after the previous task in the same operation j is complete at time point p. Constraints (5.21) and (5.22) respectively state that if an operation j starts or ends at two distinct time points, then the later time point must correspond to a later time. These constraints have proven to improve CPU time and ensure robustness and feasibility in the model. The following sequence constraints are specific to the existence of reusable water storage.

$$ts_{\text{in}}\left(s_{\text{out},j},p\right) \geq t_p\left(s_{\text{out},j},p\right) - H\left(1 - ys_{\text{in}}\left(s_{\text{out},j},p\right)\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{5.23}$$

$$ts_{\text{in}}\left(s_{\text{out},j},p\right) \leq t_p\left(s_{\text{out},j},p\right) + H\left(1 - ys_{\text{in}}\left(s_{\text{out},j},p\right)\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{5.24}$$

$$ys_{\text{in}}\left(s_{\text{out},j},p\right) \leq y\left(s_{\text{in},j},p-1\right), \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j}, p \in P, p > p_1 \tag{5.25}$$

Constraints (5.23) and (5.24) stipulate that when the water stream is transferred from operation j to reusable water storage, then the time of transfer should coincide with the completion of operation j. however, operation j will only be completed and able to transfer water to storage at time point p if it started at time point $p-1$. Also, the fact that operation j commenced at time point $p-1$ does not necessarily mean that it will transfer water to storage at time point p, since this water could be immediately reused/recycled and/or removed as effluent. This is captured by constraints (5.25). The following constraints (5.26), (5.27) and (5.28) are similar to (5.23), (5.24) and (5.25), but apply to the outlet stream of reusable water storage.

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \geq t_u\left(s_{\text{in},j},p\right) - H\left(1 - ys_{\text{out}}\left(s_{\text{in},j},p\right)\right), \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{5.26}$$

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \leq t_u\left(s_{\text{in},j},p\right) + H\left(1 - ys_{\text{out}}\left(s_{\text{in},j},p\right)\right), \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{5.27}$$

$$ys_{\text{out}}\left(s_{\text{in},j},p\right) \leq y\left(s_{\text{in},j},p\right), \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{5.28}$$

Constraints (5.26) and (5.27) state that when water stream is transferred from storage to any operation j for reuse, then the time of transfer must coincide with the start of operation j. Constraints (5.28) ensures that whenever a water stream is transferred from storage to operation j at time point p, then operation j must commence at time point p. However, operation j can start at time point p even if there is no reusable water stream transferred from storage, since water could be received from recycle/reuse and fresh water streams.

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \geq ts_{\text{out}}\left(s_{\text{in},j'},p'\right) - H\left(2 - ys_{\text{out}}\left(s_{\text{in},j},p\right) - ys_{\text{out}}\left(s_{\text{in},j'},p'\right)\right), \quad \forall j,j' \in J, s_{\text{in},j}, s_{\text{in},j'} \in S_{\text{in},j}, p,p' \in P, p \geq p' \tag{5.29}$$

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \geq ts_{\text{out}}\left(s_{\text{in},j'},p\right) - H\left(2 - ys_{\text{out}}\left(s_{\text{in},j},p\right) - ys_{\text{out}}\left(s_{\text{in},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{in},j}, s_{\text{in},j'} \in S_{\text{in},j}, p \in P \tag{5.30}$$

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \leq ts_{\text{out}}\left(s_{\text{in},j'},p\right) + H\left(2 - ys_{\text{out}}\left(s_{\text{in},j},p\right) - ys_{\text{out}}\left(s_{\text{in},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{in},j}, s_{\text{in},j'} \in S_{\text{in},j}, p \in P \tag{5.31}$$

Constraints (5.29) ensures that if reusable water is transferred from reusable water storage to operation j' at time point p' and later transferred to the same or another operation j at time point p, then the later time point must correspond to a later time. If the transfer of water from reusable water storage to different operations j and j' takes place at the same time point p, then this time point must correspond to exactly the same time as enforced by both constraints (5.30) and (5.31).

$$ts_{\text{in}}\left(s_{\text{out},j},p\right) \geq ts_{\text{in}}\left(s_{\text{out},j'},p'\right) - H\left(2 - ys_{\text{in}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p'\right)\right), \quad \forall j,j' \in J, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p,p' \in P, p \geq p' \tag{5.32}$$

$$ts_{\text{in}}\left(s_{\text{out},j},p\right) \geq ts_{\text{in}}\left(s_{\text{out},j'},p\right) - H\left(2 - ys_{\text{in}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p \in P \tag{5.33}$$

$$ts_{\text{in}}\left(s_{\text{out},j},p\right) \leq ts_{\text{in}}\left(s_{\text{out},j'},p\right) + H\left(2 - ys_{\text{in}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p \in P \tag{5.34}$$

Constraints (5.32), (5.33) and (5.34) are similar to constraints (5.29), (5.30) and (5.31), but apply to the inlet stream of reusable water storage.

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \geq ts_{\text{in}}\left(s_{\text{out},j'},p'\right) - H\left(2 - ys_{\text{out}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p'\right)\right),$$
$$\forall j,j' \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p,p' \in P, p \geq p' \tag{5.35}$$

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \geq ts_{\text{in}}\left(s_{\text{out},j'},p\right) - H\left(2 - ys_{\text{out}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p\right)\right), \tag{5.36}$$
$$\forall j,j' \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p \in P$$

$$ts_{\text{out}}\left(s_{\text{in},j},p\right) \leq ts_{\text{in}}\left(s_{\text{out},j'},p\right) + H\left(2 - ys_{\text{out}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p\right)\right), \tag{5.37}$$
$$\forall j,j' \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j}, s_{\text{out},j'} \in S_{\text{out},j}, p \in P$$

Constraints (5.35) states that if reusable water is transferred from operation j ́to storage at time point p ́ and later transferred from reusable water storage to the same or another operation j at time point p, then the latter transfer must correspond to a later time. On the other hand, if the transfer of reusable water into and out of reusable water storage occurs at the same time point p, then this should correspond to the same time, i.e. the time of transfer from operation j ́ to reusable water storage must be equal to the time of transfer from reusable water storage to operation j. This is captured by constraints (5.36) and (5.37).

5.3.3 Additional Constraints

In order to ensure that all the reusable water from storage is completely reused within the chosen time horizon, a constraints for zero storage at the end of the time horizon is formulated as shown in Constraints (5.38).

$$qs\,(p) = 0, p = |P| \tag{5.38}$$

Alternatively, constraints (5.39) might be used. This constraints ensures that the overall amount of freshwater used equals the overall amount of effluent produced over the time horizon of interest. This condition ascertains that no water remains in reusable water storage at the end of the time horizon.

$$\sum_{s_{\text{out},j}} \sum_{p} m_e\left(s_{\text{out},j},p\right) = \sum_{s_{\text{in},j}} \sum_{p} m_f\left(s_{\text{in},j},p\right), p \in P, s_{\text{out},j} \in S_{\text{out},j}, s_{\text{in},j} \in S_{\text{in},j} \tag{5.39}$$

The two sets of constraints presented in Sections 3.1 and 3.2 constitute an overall mathematical model, which is used in the proposed two-stage solution algorithm.

5.3.4 Objective Function

The objective function for the mathematical model is dependent on the stage of the two-stage solution approach. In the first stage, the objective is simply to minimize freshwater, given a fixed capacity of central reusable water storage. However, in the second stage, the objective is to minimize the capacity of reusable water storage, whilst obeying the minimum amount of freshwater obtained in the first stage. Therefore, in the second stage the capacity of central reusable water storage is no longer fixed, but treated as a variable with a known upper bound. The following section gives the detailed solution procedure.

5.4 Solution Procedure

The solution procedure for the problem stated above involves two optimisation stages in which freshwater and reusable water storage capacity are minimised sequentially, as explained in Section 5.4.1 below.

5.4.1 Two-Stage Optimisation Algorithm for Freshwater and Reusable Water Storage Minimisation

The optimisation procedure presented in this chapter entails two stages as summarized in Fig. 5.3. In the first stage, a mathematical model for minimisation of freshwater requirement is solved based on maximum potential reusable water storage, Q_s^U. For clarity, this model will be referred to as model M1 in this chapter. In the second stage, the minimum freshwater requirement obtained from model M1 is used as an input parameter in another mathematical model for which the objective function is the minimisation of reusable water storage. This model will be referred to as model M2 in this chapter. Since different amounts of reusable water will be stored at various intervals within the time horizon of interest, the minimum reusable water storage capacity will correspond to the maximum amount of reusable water stored at any point within the time horizon of interest as obtained from model M2 (Constraints (5.40)).

$$Q_s^L = \max_p \left\{ q_s^L(p) \right\} \tag{5.40}$$

All the solutions presented in the following case study were obtained using a 1.5 GHz Pentium M processor and GAMS 2.5/DICOPT. The NLP and MILP combination of solvers selected for DICOPT were CONOPT and CPLEX 7.0, respectively.

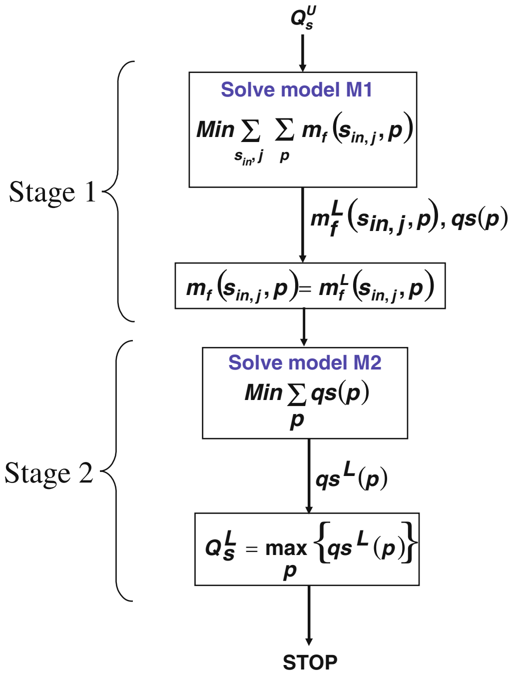


Fig. 5.3 Algorithm for reusable water storage minimisation (Majozi, 2006)

5.5 Case Study

To illustrate the application of the proposed algorithm the agrochemicals problem already presented in Chapter 4 of this textbook is revisited. It involves a completely batch operation wherein reusable water is generated from liquid–liquid extraction (product washing) operations with water as the aqueous phase in the production of three agrochemicals A, B and D. The data for the production of these products are shown in Table 5.1. These agrochemicals are produced in batch reactors. All three reactions form sodium chloride (NaCl) as a byproduct which is later removed from

Table 5.1 Data for the case study

Operation (j)	Water (kg)	C_{out}^U (j) (kg salt/kg water)	C_{in}^U (j) (kg salt/kg water)	$t_u(s,p)$ (h)	$t_p(s,p)$ (h)	$M(j)$ (kg)
A	1000	0.1	0	0	3	100
B	280	0.51	0.25	0	4	72.8
C	[300, 400]	0.1	0.1	4	5.5	0
D	280	0.51	0.25	2	6	72.8
E	[300, 400]	0.1	0.1	6	7.5	0
Total	2360					245.6

the final product. The removal of this byproduct is effected by the use of fresh water. It is worth mentioning that, although the aim of the washes is to remove NaCl, there are always traces of organics in water. In formulating the problem, however, it was assumed that the concentration of these organics is virtually negligible.

In the case of A, the reaction takes place in an organic solvent which is highly immiscible with water, so that water is required solely for washing the salt. In the case of B and D, however, water is used as the reaction solvent, and a further quantity is used for washing the product. While investigating this secondary washing of B and D, it was found that the salt load removed from the product was essentially zero due to the fact that most of it had been removed with the reaction solvent water. However, it was considered that the washing step should not be discarded, as it constituted a quality control precaution in case of unforeseen process problems. The secondary washing stages for products B and D are shown as operations C and E in Table 5.1. The foregoing explanation justifies the zero contaminant mass loads corresponding to these operations. The minimum amount of water required in the secondary washing operation is 300 kg per batch, whilst the maximum amount is 400 kg.

The timing of the reaction and washing sequences is considered to be fixed by product requirements, which implies that there is no freedom to change the sequence to optimize the use of water.

5.5.1 Computational Results

The corresponding mathematical formulation entails 5534 constraints, 1217 continuous and 280 binary variables. In solving model M1, on average 4000 nodes were explored in the branch and bound search tree. The solution required three major iterations and took 309.41 CPU seconds to obtain the optimal solution of 1285.50 kg. This corresponds to 45.53% reduction in freshwater demand compared to the amount of water required in a situation in which there is no reuse, i.e. 2360 kg. A water reuse/recycle network that corresponds to this solution is shown in Fig. 5.4 and the Gantt chart in Fig. 5.5. As evident from the Gantt chart, storage is necessitated by the time gaps between processes A and C as well processes A and E. It is worth mentioning, however, that global optimality of this optimal value cannot be proven due to nonconvexity of the mathematical model at hand. A better optimum might exist.

Once the minimum water requirement has been obtained, it is then used as an input parameter in the determination of minimum reusable water storage, i.e. stage 2 of the algorithm. Figure 5.6 shows the water reuse network with minimum reusable water storage, without compromising the minimum freshwater requirement. The corresponding Gantt chart is shown in Fig. 5.7. This shows a reduction of 500 kg from the initial 800 kg used in stage 1 of the algorithm, i.e. 62.5% reduction in capacity. The major reduction in storage capacity is due the recycle water stream from operation C back to storage. This water stream is later reused in operation E,

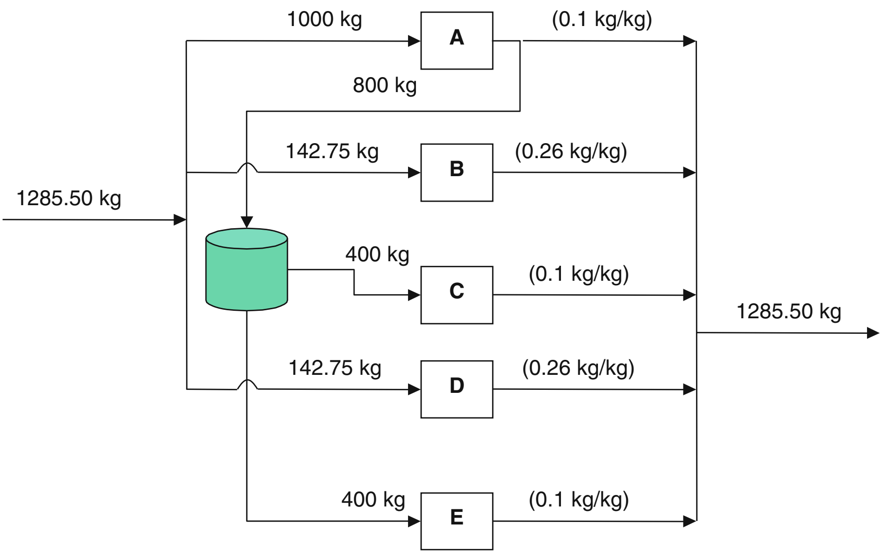
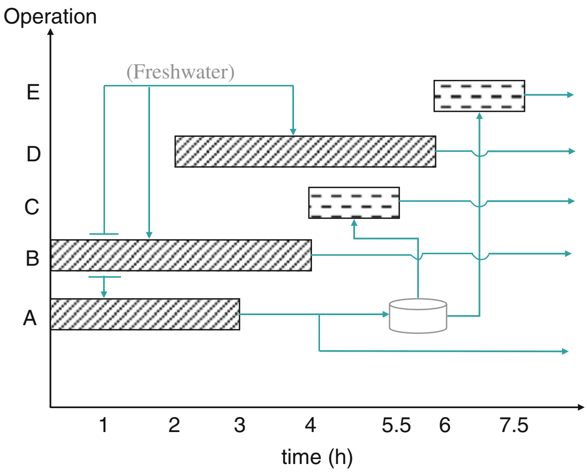


Fig. 5.4 Water reuse/recycle network for minimum water use – model M1 (Majozi, 2006)

Fig. 5.5 Gantt chart corresponding to minimum water use – model M1 (Majozi, 2006)



since its concentration allows. Model M2 involved two major iterations and was solved in only 1.51 CPU seconds due to the reduction in the number of variables. The reusable water storage profiles for model M1 and model M2 are shown Figs. 5.8a, b, respectively.

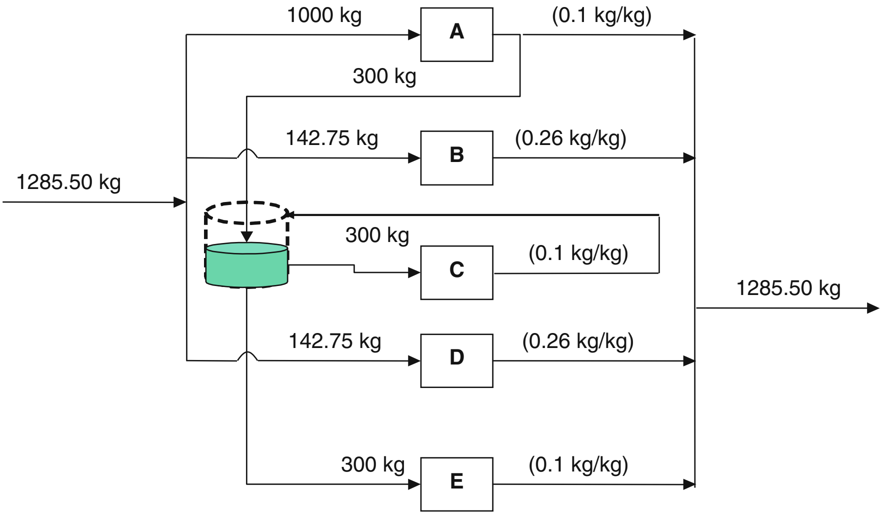


Fig. 5.6 Water reuse/recycle network for minimum reusable water storage – model M2 (Majozi, 2006)

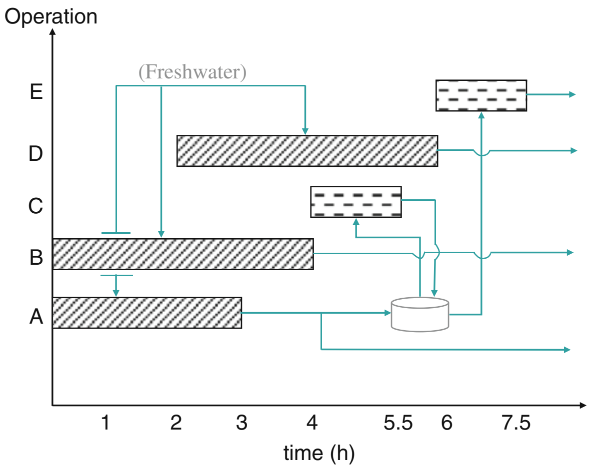


Fig. 5.7 Gantt chart corresponding to minimum reusable water storage – model M2 (Majozi, 2006)

5.6 Concluding Remarks

An algorithm for the minimisation of reusable water storage in batch facilities has been developed. The algorithm involves a two-stage approach. In the first stage the objective is to minimize freshwater requirement, given an upper bound on reusable water storage. Once the minimum water has been determined, it is then fixed and used as an input parameter in the second stage of which the objective is the minimization of storage. This procedure ensures that storage is minimized without

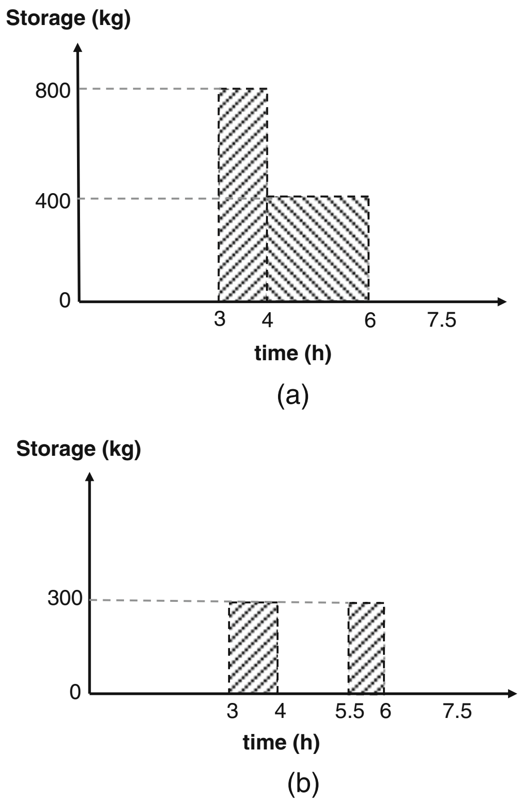


Fig. 5.8 Storage profiles for (**a**) model M1 and (**b**) model M2 (Majozi, 2006)

compromising on freshwater demand. Application of this procedure to a simple case study has shown more than 45% reduction in freshwater demand and 62.5% reduction in reusable water storage requirement. Since the mathematical formulation is based on a continuous-time framework, the problem was solved within a reasonable CPU time, i.e. 309.41 CPU seconds for the first stage and 1.51 CPU seconds for the second stage.

5.7 Exercise

Task: Apply the algorithm presented in this chapter to the problem given in Table 5.2. This is indeed similar to the problem already presented in Chapter 4 and revisited in this chapter, except that only duration is given instead of prescribed starting and ending times. An added necessary condition that has to be considered is that operation C has to commence immediately after operation B and operation E has to commence immediately after operation D and not vice versa. Assume a time horizon of 12 h.

Table 5.2 Data for the exercise

Operation (j)	Water (kg)	C_{out}^{U} (j) (kg salt/kg water)	C_{in}^{U} (j) (kg salt/kg water)	Duration (h)	$M(j)$ (kg)
A	1000	0.1	0	3	100
B	280	0.51	0.25	4	72.8
C	[300, 400]	0.1	0.1	1.5	0
D	280	0.51	0.25	4	72.8
E	[300, 400]	0.1	0.1	1.5	0
Total	2360				245.6

References

Majozi, T., 2005. Wastewater minimization using central reusable water storage in batch processes. Comput. Chem. Eng., 29(7): 1631–1646.

Majozi, T., 2006. Storage design for maximum wastewater reuse in multipurpose batch plants. Ind. Eng. Chem. Res., 45(17): 5936–5943.

Majozi, T., Zhu, X.X., 2001. A novel continuous time MILP formulation for multipurpose batch plants. 1. Short-term scheduling. Ind. Eng. Chem. Res., 40(25): 5935–5949.

Chapter 6
Wastewater Minimisation in *Multipurpose* Batch Plants: *Multiple* Contaminants

Overview This chapter presents the generalised version of the mathematical formulation presented in Chapter 4. The presented formulation is for wastewater minimisation in multipurpose batch processes where there are multiple contaminants present in the system. The reader is reminded that Chapter 4 was only applicable to multiproduct batch plants characterised by single contaminants. Moreover, only water using operations were considered in the analysis in exclusion of production scheduling that might accompany wastewater minimisation. In this chapter, a combined production scheduling and wastewater minimisation framework is presented and applied to demonstrate its capability. More succinctly, the problems of production scheduling and wastewater minimisation in multipurpose batch plants are solved simultaneously. Needless to mention, this still remains one of the most advanced problems ever addressed in batch process integration. The first section of this chapter deals with some background to the problem. The formal problem statement is given in the second section. The third section is centred on the mathematical formulation, whilst the fourth section describes the solution procedure. The application of the methodology to a number of illustrative examples is given in the fifth section and the conclusions are given in the sixth section.

6.1 Multiple Contaminant Wastewater Minimisation Background

Wastewater contaminated with multiple contaminants is a common occurrence in any industrial facility. Past wastewater minimisation methodologies have been focussed on single contaminant systems. This restricts the application of the methodologies to a small range of problems. Therefore the development of a multiple contaminant wastewater methodology is mandatory since it finds broader industrial application.

The methodology presented in this chapter is derived for two distinct cases. In the first case there is no central storage vessel available for wastewater storage. In this

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_6,

case only direct recycle/reuse is possible. In the second case there is a central storage vessel for wastewater storage. In this case reuse either occurs directly between two units or between two units via the central storage vessel, which constitutes indirect recycle/reuse. As noted in the previous chapter the usage of storage allows for greater reuse possibilities, since it affords the bypassing of timing constraints that are inherent in batch processes.

The single contaminant wastewater minimisation methodology presented by Majozi (2005) forms the basis of the multiple contaminant wastewater minimisation methodology. The resulting methodology thus determines the minimum wastewater flowrate and the corresponding schedule simultaneously. Furthermore, the underlying scheduling framework is based on the uneven discretization of the time horizon and the state sequence network (SSN) representation of a process (Majozi and Zhu, 2001). In batch operation terms, the operational policy adopted is the finite intermediate storage policy, in which material is transferred to storage after the required processing has taken place. However, the availability of storage is not guaranteed, since it has a definite upper bound. The problem addressed is described in the following section.

6.2 Problem Statement

The problem addressed in this chapter can be formally stated as follows.

It is given:

(i) the contaminant mass load of each contaminant,
(ii) the necessary cost and stoichiometric data,
(iii) the maximum inlet and outlet concentrations of each contaminant,
(iv) the available units and their capacities,
(v) the time horizon of interest and
(vi) the maximum storage available for water reuse.

The objective is then to determine the production schedule that generates the least amount of effluent through the exploitation of wastewater recycle/reuse. It should be emphasised that recycle in this context refers to wastewater being used by the same unit from which it was produced and reuse refers to the usage of wastewater in a different unit to which it was produced.

6.3 Mathematical Formulation

The mathematical formulation presented in this chapter comprises the following sets, variables and parameters.

Sets

$P \;=\; \{p \,|p \;=\text{time point}\}$

$J \;=\; \{j \,|j = \text{unit}\}$

$C \;=\; \{c \,|c \;=\text{contaminant}\}$

$S_{\text{in}} \;=\; \{S_{\text{in}} \,|S_{\text{in}} \;=\text{input state into any unit}\}$

$S_{\text{out}} \;=\; \{S_{\text{out}} \,|S_{\text{out}} =\text{output state from any unit}\}$

$S \;=\; \{s \,|s \;=\text{any state}\} = S_{\text{in}} \cup S_{\text{out}}$

$S_{\text{in},j} \;=\; \left\{S_{\text{in},j} \,\middle|S_{\text{in},j} \;=\text{input state into unit}\, j\right\} \subseteq S_{\text{in}}$

$S^{*}_{\text{in},j} \;=\; \left\{S^{*}_{\text{in},j} \,\middle|S^{*}_{\text{in},j}\;\text{input state into unit}\, j\right\} \subseteq S_{\text{in},j}$

$S_{\text{out},j} = \; \left\{S_{\text{out},j} \,\middle|S_{\text{out},j} \;=\text{output state from unit}\, j\right\} \subseteq S_{\text{out}}$

Variables Associated with Wastewater Minimization

$mw_{\text{in}}\left(s_{\text{out},j}, p\right)$	mass of water into unit j for cleaning state s_{out} at time point p
$mw_{\text{out}}\left(s_{\text{out},j}, p\right)$	mass water produced at time point p from unit j
$mw_{f}\left(s_{\text{out},j}, p\right)$	mass of fresh water into unit j at time point p
$mw_{e}\left(s_{\text{out},j}, p\right)$	mass of effluent water from unit j at time point p
$mw_{r}\left(s_{\text{out},j}, s_{\text{out},j'}, p\right)$	mass of water recycled to unit j' from j at time point p
$ms_{\text{in}}\left(s_{\text{out},j}, p\right)$	mass of water to storage from unit j at time point p
$ms_{\text{out}}\left(s_{\text{out},j}, p\right)$	mass of water from storage to unit j at time point p
$c_{\text{in}}\left(s_{\text{out},j}, c, p\right)$	inlet concentration of contaminant c, unit j, time point p
$c_{\text{out}}\left(s_{\text{out},j}, c, p\right)$	outlet concentration of contaminant c, unit j, time point p
$cs_{\text{in}}\,(c, p)$	inlet concentration of contaminant c into storage at time point p
$cs_{\text{out}}\,(c, p)$	outlet concentration of contaminant c from storage at time point p
$tw_{\text{in}}\left(s_{\text{out},j}, p\right)$	time that the water is used at time point p in unit j
$tw_{\text{out}}\left(s_{\text{out},j}, p\right)$	time at which the water is produced at time point p from unit j
$tw_{r}\left(s_{\text{out},j}, s_{\text{out},j'}, p\right)$	time at which water is recycled from unit j to unit j' at time point p
$yw\left(s_{\text{out},j}, p\right)$	binary variable showing usage of unit j at time point p
$yw_{r}\left(s_{\text{out},j}, s_{\text{out},j'}, p\right)$	binary variable showing usage of recycle from unit j to unit j' at time point p

$ys_{in}\left(s_{out,j},p\right)$	binary variable showing usage of water into storage from unit j at time point p
$ys_{out}\left(s_{out,j},p\right)$	binary variable showing usage of water from storage to unit j at time point p
$ts_{in}\left(s_{out,j},p\right)$	time at which water goes to storage from unit j at time point p
$ts_{out}\left(s_{out,j},p\right)$	time at which water leaves storage to unit j at time point p

Variables Associated with Production Scheduling

$t_{out}\left(s_{out,j},p\right)$	time at which a state is produced from unit j at time point p
$t_{in}\left(s_{in,j},p\right)$	time at which a state is used in or enters unit j at time point p
$q_s\,(s,p)$	amount of state s stored at time point p
$m_{out}\left(s_{out,j},p\right)$	amount of state produced from unit j at time point p
$m_{in}\left(s_{in,j},p\right)$	amount of state used in or enters unit j at time point p
$y\left(s^*_{in,j},p\right)$	binary variable associated with usage of state s at time point p
$d\,(s_{out},p)$	amount of state delivered to customers at time point p

Parameters Associated with Wastewater Minimization

$\Psi\,(j)$	mass production factor (kg raw material/kg water)
$SP\,(j)$	selling price of product from unit j (c.u./kg product)
$CR\,(j)$	cost of raw material going into unit j (c.u./kg raw material)
CE	cost of effluent water treatment (c.u./kg water)
CF	cost of freshwater (c.u./kg water)
$M\left(s_{out,j},c\right)$	mass load of contaminant c added from unit j to the water stream
$Mw^U\left(s_{out,j}\right)$	maximum inlet water mass of unit j
$C^U_{in}\left(s_{out,j},c\right)$	maximum inlet concentration of contaminant c in unit j
$C^U_{out}\left(s_{out,j},c\right)$	maximum outlet concentration of contaminant c from unit j
$\tau w\left(s_{out,j}\right)$	mean processing time of unit j
Qw^o_s	initial amount of water stored in the storage vessel
Qw^U_s	maximum storage capacity of the water storage vessel
$CS^0_{out}\,(c)$	initial concentration of contaminant c in the storage vessel

Parameters Associated with Production Scheduling

V^U_j	maximum design capacity of a particular unit j
V^L_j	minimum design capacity of a particular unit j
H	time horizon of interest

$\tau\left(s^{*}_{\text{in},j}\right)$	mean processing time for a state
$Q^{0}_{s}(s)$	initial amount of state s stored
$Q^{U}_{s}(s)$	maximum amount of state s stored within the time horizon of interest
$CP(s)$	Selling price of product s, s = product

The constraints considered in the mathematical formulation are divided into two modules. The first deals with the mass balance constraints and the second with the sequencing and scheduling constraints. The mass balance constraints for the case where there is no central storage are slightly different to those for the case where there is. The mass balances for each are described in the mass balance module below. The sequencing and scheduling module will be described, for both cases, in a subsequent section. The nomenclature for all the sets, variables and parameters can be found in the nomenclature list.

6.3.1 Mass Balance Constraints

Mass Balance Constraints Without Storage

The mathematical formulation for the multiple contaminant system without storage is based on the superstructure given in Fig. 6.1. The superstructure is very similar to that used by Majozi (2005), but embraces the concept of multiple contaminants. Figure 6.1a shows an individual water using unit. From the figure one can see that water entering a unit comprises of freshwater and directly recycled/reused water. This is similar for water leaving a unit, where this water is either discarded or recycled/reused. Figure 6.1b is the overall plant superstructure. Figure 6.1a is, in essence, a magnification of Fig. 6.1b.

Water and contaminant balances are the first mass balance constraints considered. Constraints (6.1) is an inlet water balance. The amount of water entering a unit at a time point is the amount of freshwater entering the unit and the sum of water recycled/reused to that unit. Water leaving a unit at a time point comprises of water discarded as effluent and water recycled/reused. This is given in constraints (6.2). It is assumed that water is not generated or consumed within an operation, therefore, the amount of water entering a unit is equal to the amount of water leaving a unit, as given in constraints (6.3). However, the possibility of water generation or consumption within an operation can easily be included with the addition of terms accounting for this in constraints (6.3). Constraints (6.4) is the definition of inlet concentration. The inlet concentration of contaminant c at a time point is the ratio of the mass of contaminant c entering the unit with recycled/reused water to the total amount of water entering the unit. It is important to note that the inlet concentration is defined for each contaminant present in the system. Constraints (6.5) is a contaminant balance over a unit. Constraints (6.5) states that the mass of contaminant c leaving the unit comprises the contaminant mass that entered the unit and the contaminant mass load added to water due to the operation of the unit.

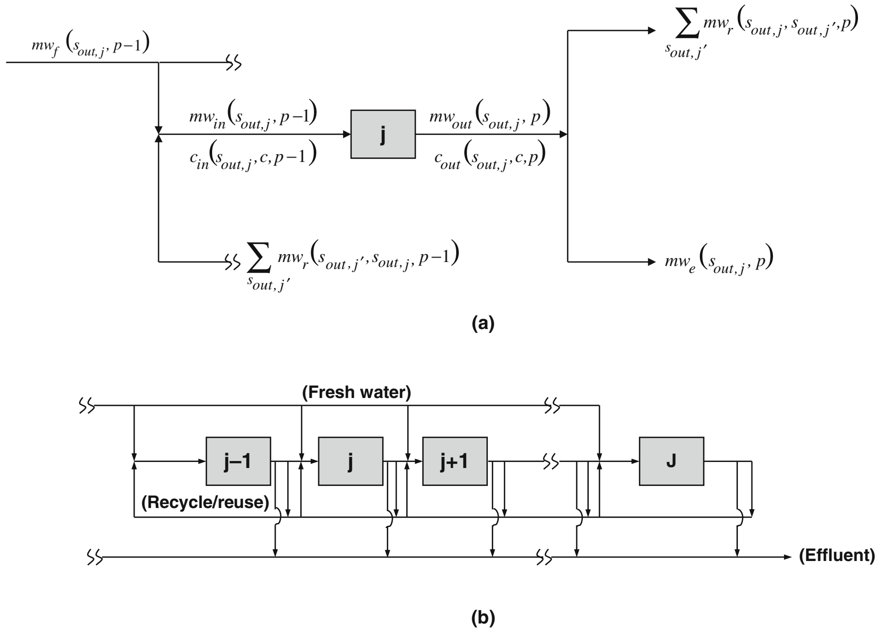


Fig. 6.1 Superstructure used for multiple contaminant case with no central wastewater storage (Majozi and Gouws, 2009)

$$mw_{\text{in}}\left(s_{\text{out},j},p\right)=\sum_{s_{\text{out},j'}} mw_r\left(s_{\text{out},j'},s_{\text{out},j},p\right)+mw_f\left(s_{\text{out},j},p\right),\quad \forall j,j'\in J, s_{\text{out},j}\in S_{\text{out},j}, p\in P \tag{6.1}$$

$$mw_{\text{in}}\left(s_{\text{out},j},p-1\right)=mw_{\text{out}}\left(s_{\text{out},j},p\right),\quad \forall j\in J, s_{\text{out},j}\in S_{\text{out},j}, p\in P \tag{6.2}$$

$$mw_{\text{out}}\left(s_{\text{out},j},p\right)=\sum_{s_{\text{out},j'}} mw_r\left(s_{\text{out},j},s_{\text{out},j'},p\right)+mw_e\left(s_{\text{out},j},p\right),\quad \forall j,j'\in J, s_{\text{out},j}\in S_{\text{out},j}, p\in P \tag{6.3}$$

$$\begin{aligned}mw_{\text{out}}\left(s_{\text{out},j},p\right)c_{\text{out}}\left(s_{\text{out},j},c,p\right)=\;&mw_{\text{in}}\left(s_{\text{out},j},p-1\right)c_{\text{in}}\left(s_{\text{out},j},c,p-1\right)\\&+M(s_{\text{out}},c)yw\left(s_{\text{out},j},p-1\right),\\&\forall j\in J, s_{\text{out},j}\in S_{\text{out},j}, p\in P, p>p_1, c\in C\end{aligned} \tag{6.4}$$

$$c_{\text{in}}\left(s_{\text{out},j},c,p\right)=\frac{\sum_{s_{\text{out},j'}} mw_r\left(s_{\text{out},j'},s_{\text{out},j},p\right)c_{\text{out}}\left(s_{\text{out},j'},c,p\right)}{mw_{\text{in}}\left(s_{\text{out},j},p\right)},\quad \forall j,j'\in J, s_{\text{out},j'}\in S_{\text{out},j}, p\in P, c\in C \tag{6.5}$$

The outlet concentration of each contaminant c in unit j cannot exceed its maximum limit as stated in constraints (6.6). Constraints (6.7) ensures that the total water into a unit j does not exceed the maximum allowable for the operation in unit j. Constraints (6.8) restricts the mass of water recycled into the unit j to the maximum allowable water for the operation in unit j. Constraints (6.9) stipulates that the inlet concentration for contaminant c into unit j cannot exceed its upper limit.

$$c_{\text{out}}\left(s_{\text{out},j},c,p\right) \leq C_{\text{out}}^{U}\left(s_{\text{out},j},c\right) yw\left(s_{\text{out},j},p-1\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P, p > p_1, c \in C \tag{6.6}$$

$$mw_{\text{in}}\left(s_{\text{out},j},p\right) \leq Mw^{U}\left(s_{\text{out},j}\right) yw\left(s_{\text{out},j},p\right), \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.7}$$

$$mw_{r}\left(s_{\text{out},j'},s_{\text{out},j},p\right) \leq Mw^{U}\left(s_{\text{out},j}\right) yw_{r}\left(s_{\text{out},j'},s_{\text{out},j},p\right), \quad \forall j,j' \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.8}$$

$$c_{\text{in}}\left(s_{\text{out},j},c,p\right) \leq C_{\text{in}}^{U}(s_{\text{out},j},c) yw\left(s_{\text{out},j},p\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P, c \in C \tag{6.9}$$

The maximum amount of water used by a unit is determined using constraints (6.10). This amount is used as the limit in constraints (6.8) and (6.9). In a multiple contaminant system there exists a limiting component for each operation in each unit. The limiting component is the component that requires the largest amount of water to remove the required mass load and still comply with the maximum inlet and outlet concentrations. For a certain operation in a certain unit there could exist multiple limiting components, however, the amount of water required by each will be the same. It is important to note that when the maximum amount of water is used, the concentration of the non-limiting components will be below their respective maxima.

$$Mw^{U}\left(s_{\text{out},j}\right) = \max_{c \in C}\left\{\frac{M\left(s_{\text{out},j},c\right)}{C_{\text{out}}^{U}\left(s_{\text{out},j},c\right) - C_{\text{in}}^{U}\left(s_{\text{out},j},c\right)}\right\}, \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j} \tag{6.10}$$

The above mass balance constraints suffice for the case where there is no intermediate storage available for wastewater. The mass balance constraints necessary for the case where there is a central storage vessel available for wastewater are presented in the following section.

Mass Balance Constraints Including Central Storage

The formulation for the case where there is wastewater storage available is based on the superstructure given in Fig. 6.2. This superstructure is similar to the previous, however, there is a central storage vessel available. Once again Fig. 6.2a shows an individual water using unit and Fig. 6.2b shows the overall plant superstructure.

Mass balance constraints (6.1), (6.3) and (6.5) need to be reformulated to account for the water from storage. The water into a unit in this case is not only comprised of freshwater and directly recycled/reused water, but also water from storage. This

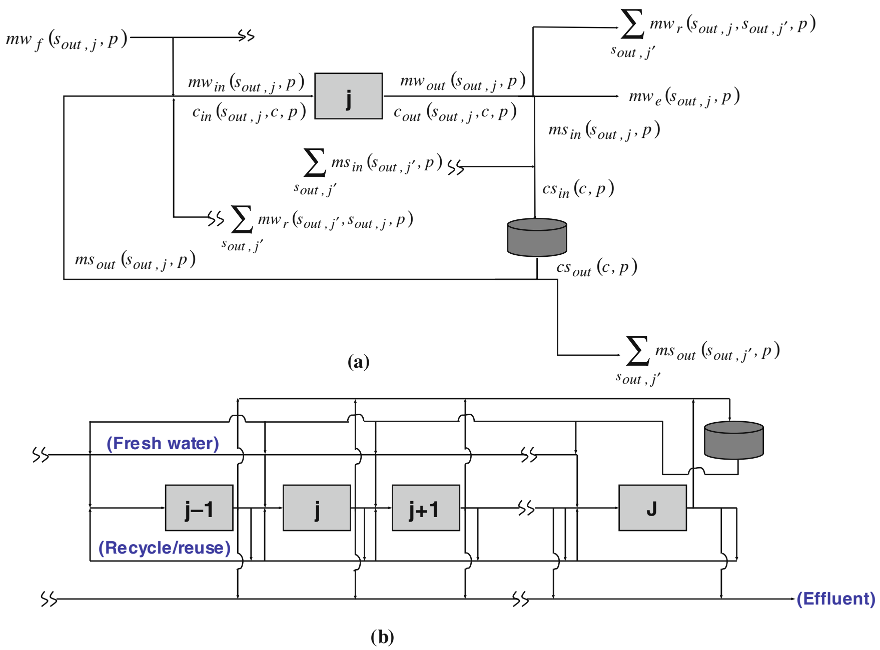


Fig. 6.2 Superstructure for multiple contaminant methodology including central storage (Majozi and Gouws, 2009)

is captured in constraints (6.11). Water leaving a unit in this case cannot only be discarded as effluent or directly recycled/reused, but also sent to storage for reuse at a later stage, given in constraints (6.12). The inlet concentration defined in constraints (6.5) is reformulated since the inlet concentration is now dependent not only on contaminant mass from directly recycled/reused water, but contaminant mass in water from storage. The inlet concentration where there is wastewater storage present is given in constraints (6.13).

$$mw_{\text{in}}\left(s_{\text{out},j},p\right) = \sum_{s_{\text{out},j'}} mw_r\left(s_{\text{out},j'}, s_{\text{out},j}, p\right) + mw_f\left(s_{\text{out},j},p\right) + ms_{\text{out}}\left(s_{\text{out},j},p\right),$$
$$\forall j,j' \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \quad (6.11)$$

$$mw_{\text{out}}\left(s_{\text{out},j},p\right) = \sum_{s_{\text{out},j'}} mw_r\left(s_{\text{out},j}, s_{\text{out},j'}, p\right) + mw_e\left(s_{\text{out},j},p\right) + ms_{\text{in}}\left(s_{\text{out},j},p\right),$$
$$\forall j,j' \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \quad (6.12)$$

$$mw_{\text{in}}\left(s_{\text{out},j},p\right)c_{\text{in}}\left(s_{\text{out},j},c,p\right)=\sum_{s_{\text{out},j'}} mw_r\left(s_{\text{out},j'},s_{\text{out},j},p\right)c_{\text{out}}\left(s_{\text{out},j'},c,p\right)$$
$$+\,ms_{\text{out}}\left(s_{\text{out},j},p\right)cs_{\text{out}}\left(c,p\right),$$
$$\forall j,j'\in J,s_{\text{out},j'}\in S_{\text{out},j},p\in P,c\in C \quad (6.13)$$

Due to the presence of the central storage vessel, mass balance constraints around the vessel also need to be accounted for. These constraints pertain to water entering and leaving the vessel and the amount of water stored in the vessel. The first of these constraints is a water balance over the vessel, as given in constraints (6.14). The amount of water stored at any time point is the amount stored at the previous time point and the difference between the amount of water entering and leaving the storage vessel at that time point. Constraints (6.15) is the water balance at the first time point within the time horizon. This constraints states that the amount of water stored at the first time point is the difference between the initial amount of water within the vessel and the amount of water leaving the vessel. Constraints (6.16) ensures that the amount of water stored within the storage vessel does not exceed the capacity of the storage vessel.

$$qw_s\left(p\right)=qw_s\left(p-1\right)+\sum_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j},p\right)-\sum_{s_{\text{out},j}} ms_{\text{out}}\left(s_{\text{out},j},p\right),$$
$$\forall j\in J,s_{\text{out},j}\in S_{\text{out},j},p\in P,p>p_1 \quad (6.14)$$

$$qw_s\left(p_1\right)=Qw_s^0-\sum_{s_{\text{out},j}} ms_{\text{out}}\left(s_{\text{out},j},p_1\right),\ \forall j\in J,s_{\text{out},j}\in S_{\text{out},j} \quad (6.15)$$

$$qw_s\left(p\right)\le Qw_s^U,\ \forall p\in P \quad (6.16)$$

The inlet concentration into a storage vessel is defined in constraints (6.17). The concentration within the vessel is defined in constraints (6.18). It is assumed in constraints (6.18) that the storage vessel is perfectly mixed, hence the uniform concentration throughout the vessel. The initial concentration within the storage vessel is given in constraints (6.19). Constraints (6.20) ensures that the amount of water used from the storage vessel is less than the maximum amount allowable in the receiving unit. It is important to note that the contaminant balances presented below hold for each contaminant present within the system.

$$cs_{\text{in}}\left(c,p\right)=\frac{\sum\limits_{s_{\text{out},j}}\left(ms_{\text{in}}\left(s_{\text{out},j},p\right)c_{\text{out}}(s_{\text{out},j},c,p)\right)}{\sum\limits_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j},p\right)}, \quad (6.17)$$
$$\forall j\in J,s_{\text{out},j}\in S_{\text{out},j},p\in P,c\in C$$

$$cs_{\text{out}}(c,p) = \frac{qw_s(p-1)\,cs_{\text{out}}(c,p-1) + \left(\sum\limits_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j},p\right)\right) cs_{\text{in}}(c,p)}{qw_s(p-1) + \sum\limits_{s_{\text{out},j}} ms_{\text{in}}\left(s_{\text{out},j},p\right)},$$
$$\forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P, p > p_1, c \in C \tag{6.18}$$

$$cs_{\text{out}}(c,p_1) = CS^0_{\text{out}}(c), \quad \forall\, c \in C \tag{6.19}$$

$$ms_{\text{out}}\left(s_{\text{out},j},p\right) \le Mw^U\left(s_{\text{out},j}\right) ys_{\text{out}}\left(s_{\text{out},j},p\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.20}$$

The mass balance constraints given above would suffice if the process were continuous. However, due to the fact that the processes dealt with are batch processes, additional constraints are required to capture the discontinuous nature of the process. This implies that the time related constraints are necessary.

6.3.2 Sequencing and Scheduling Constraints

The sequencing and scheduling constraints, for both the cases where there is a central storage vessel and where there is none, are presented below. The scheduling constraints used in the case where there is central storage comprise of those for the case where there is no storage and additional constraints that deal solely with the scheduling of the storage vessel. The sequencing and scheduling constraints can be divided into a number of groups. The first group comprises of those constraints pertaining to production sequencing and scheduling. The second group comprises of constraints pertinent to direct recycle/reuse of wastewater, whilst the third group comprises of the constraints necessary for the storage sequencing and scheduling. The final group comprises of time horizon constraints and feasibility constraints. Each group is discussed below.

Sequencing Constraints Associated with Production Scheduling

The production scheduling model has been presented in detail in Chapter 2 of this textbook, but is briefly presented in this section of the chapter for purposes of continuity and facilitation of understanding. For a detailed explanation of each of the production scheduling constraints, the reader is referred to Chapter 2.

Capacity Constraints

$$V_j^L y\left(s^*_{\text{in},j},p\right) \le \sum_{s_{\text{in},j}} m_{\text{in}}\left(s_{\text{in},j},p\right) \le V_j^U y\left(s^*_{\text{in},j},p\right),$$
$$\forall j \in J, p \in P, s_{\text{in},j} \in S_{\text{in},j}, s^*_{\text{in},j} \in S^*_{\text{in},j} \tag{6.21}$$

Material Balances

The material balances ensure the conservation of mass around each unit and concerning each state involved in production scheduling.

$$\sum_{s_{\text{in},j}} m_{\text{in}}\left(s_{\text{in},j}, p-1\right) = \sum_{s_{\text{out},j}} m_{\text{out}}\left(s_{\text{out},j}, p\right), \forall p \in P, p > p_1, j \in J, \\ \forall s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j} \tag{6.22}$$

$$q_s\left(s, p_1\right) = Q_s^0\left(s\right) - m_{\text{in}}\left(s, p_1\right), s \neq \text{product}, \forall s \in S \tag{6.23}$$

$$q_s\left(s, p\right) = q_s\left(s, p-1\right) - m_{\text{in}}\left(s, p\right), s = \text{feed}, \forall s \in S, \forall p \in P, p > p_1 \tag{6.24}$$

$$q_s\left(s, p\right) = q_s\left(s, p-1\right) + m_{\text{out}}\left(s, p\right) - m_{\text{in}}\left(s, p\right), s \neq \text{product,feed}, \forall s \in S \\ \forall p \in P, p > p_1 \tag{6.25}$$

$$q_s\left(s, p_1\right) = Q_s^0\left(s\right) - d\left(s, p_1\right), s = \text{product}, \forall s \in S \tag{6.26}$$

$$q_s\left(s, p\right) = q\left(s, p-1\right) + m_{\text{out}}\left(s, p\right) - d\left(s, p\right), s = \text{product,byproduct}, \forall s \in S \\ \forall p \in P, p > p_1 \tag{6.27}$$

Duration Constraints

The duration constraints constitutes one of the most crucial constraints as it addresses the intrinsic aspect of time in batch plants. It simply states that the time at which a particular state is produced is dependent on the duration of task that produces the same state as follows.

$$t_{\text{out}}\left(s_{\text{out},j}, p\right) = t_{\text{in}}\left(s^*_{\text{in},j}, p-1\right) + \tau\left(s^*_{\text{in},j}\right) y\left(s^*_{\text{in},j}, p-1\right), \\ \forall j \in J, p \in P, p > p_1, s^*_{\text{in},j} \in S^*_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j} \tag{6.28}$$

Assignment Constraints

The assignment constraints ensures that only one task takes place in a given unit at a given point in time.

$$\sum_{s^*_{\text{in},j}} y\left(s^*_{\text{in},j}, p\right) \leq 1, \forall p \in P, j \in J, s^*_{\text{in},j} \in S^*_{\text{in},j} \tag{6.29}$$

Storage Constraints

The storage constraints is, in essence, the extension of the mass balances. It ensures that the amount of a particular state that is stored at any point in time during the time horizon of interest does not exceed the maximum allowed.

$$q_s(s,p) \le Q^U(s), \forall s \in S, p \in P \tag{6.30}$$

Sequencing Constraints for Recycle/Reuse in the Absence of Reusable Water Storage

Scheduling of the recycle/reuse streams is important because of the discontinuous manner of the streams existence. Wastewater can only be recycled/reused if the unit producing the wastewater and the unit receiving the wastewater finish operating and begin operating at the same time, respectively. This basically means that wastewater can only be recycled/reused once it has been produced. Furthermore wastewater can only be recycled/reused if there is a unit that starts operating at the time when water is available. The recycle/reuse of wastewater to a unit is also subject to concentration constraints. The first constraints considered is constraints (6.31). This constraints stipulates that recycle/reuse between units can only take place when the unit receiving the wastewater is operating at that time point. The unit receiving the wastewater, however, does not need to receive wastewater to operate, i.e. it can operate independent of the wastewater recycle/reuse. Constraints (6.32) and (6.33) ensure that the time at which recycle/reuse of water takes place is the same time as that at which water is produced. Constraints (6.34) and (6.35) ensure that the starting time of the unit receiving water coincides with the time at which that water is recycled/reused. It is worthy to note that in the absence of recycle/reuse, constraints (6.32), (6.33), (6.34) and (6.35) become redundant.

$$yw_r\left(s_{\text{out},j}, s_{\text{out},j'}, p\right) \le yw\left(s_{\text{out},j}{}', p\right), \ \forall j,j' \in J, s_{\text{out},j} \in S_{\text{out},j} p \in P \tag{6.31}$$

$$tw_r\left(s_{\text{out},j}, s_{\text{out},j'}, p\right) \le tw_{\text{out}}\left(s_{\text{out},j}, p\right) + H\left(1 - yw_r\left(s_{\text{out},j}, s_{\text{out},j'}, p\right)\right), \quad \forall j,j' \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.32}$$

$$tw_r\left(s_{\text{out},j}, s_{\text{out},j'}, p\right) \ge tw_{\text{out}}\left(s_{\text{out},j}, p\right) - H\left(1 - yw_r\left(s_{\text{out},j}, s_{\text{out},j'}, p\right)\right), \quad \forall j,j' \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.33}$$

$$tw_r\left(s_{\text{out},j}, s_{\text{out},j'}, p\right) \le tw_{\text{in}}\left(s_{\text{out},j'}, p\right) + H\left(1 - yw_r\left(s_{\text{out},j}, s_{\text{out},j'}, p\right)\right), \quad \forall j,j' \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.34}$$

$$tw_r\left(s_{\text{out},j}, s_{\text{out},j'}, p\right) \geq tw_{\text{in}}\left(s_{\text{out},j}, p\right) - H\left(1 - yw_r\left(s_{\text{out},j}, s_{\text{out},j'}, p\right)\right), \quad \forall j, j' \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.35}$$

Sequencing and Scheduling Constraints Associated with Storage

The first constraints considered for the sequencing and scheduling of the central storage vessel is constraints (6.36). Constraints (6.36) ensures that a unit sending water to the central storage vessel has operated at the previous time point. It is important to note that the time point at which wastewater is produced is directly after the time point in which the unit starts operating. Constraints (6.36) also allows a unit to operate without sending wastewater to the central storage vessel. Constraints (6.37) and (6.38) ensure that the time at which wastewater goes to storage coincides with the time at which the wastewater is produced.

$$ys_{\text{in}}\left(s_{\text{out},j}, p\right) \leq yw\left(s_{\text{out},j}, p-1\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P, p > p_1 \tag{6.36}$$

$$ts_{\text{in}}\left(s_{\text{out},j}, p\right) \geq tw_{\text{out}}\left(s_{\text{out},j}, p\right) - H\left(1 - ys_{\text{in}}\left(s_{\text{out},j}, p\right)\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P, p > p_1 \tag{6.37}$$

$$ts_{\text{in}}\left(s_{\text{out},j}, p\right) \leq tw_{\text{out}}\left(s_{\text{out},j}, p\right) + H\left(1 - ys_{\text{in}}\left(s_{\text{out},j}, p\right)\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P, p > p_1 \tag{6.38}$$

Constraints (6.39) states that a unit can only receive water from storage if the unit is operating. The constraints also states that a unit can operate without receiving water from the storage vessel. Constraints (6.40) and (6.41) ensure that the time at which water leaves the storage vessel to a unit coincides with the time at which the unit starts operating.

$$ys_{\text{out}}\left(s_{\text{out},j}, p\right) \leq yw\left(s_{\text{out},j}, p\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.39}$$

$$ts_{\text{out}}\left(s_{\text{out},j}, p\right) \geq tw_{\text{in}}\left(s_{\text{in},j}, p\right) - H\left(1 - ys_{\text{out}}\left(s_{\text{out},j}, p\right)\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.40}$$

$$ts_{\text{out}}\left(s_{\text{out},j}, p\right) \leq tw_{\text{in}}\left(s_{\text{in},j}, p\right) + H\left(1 - ys_{\text{out}}\left(s_{\text{out},j}, p\right)\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.41}$$

Constraints (6.42), (6.43) and (6.44) deal with the scheduling aspects of two streams leaving the storage vessel. Constraints (6.42) ensures that streams leaving the storage vessel at later time points correspond to a later absolute time within the time horizon. Constraints (6.43) and (6.44) ensure that if two water streams are leaving the storage vessel at the same time point, both streams leave at the same time in the time horizon.

$$ts_{out}\left(s_{out,j},p\right) \geq ts_{out}\left(s_{out,j'},p'\right) - H\begin{pmatrix} 2 - ys_{out}\left(s_{out,j},p\right) \\ -ys_{out}\left(s_{out,j'},p'\right) \end{pmatrix}, \quad \forall j,j' \in J, s_{out,j} \in S_{out,j}, p,p' \in P, p \geq p' \tag{6.42}$$

$$ts_{out}\left(s_{out,j},p\right) \geq ts_{out}\left(s_{out,j'},p\right) - H\left(2 - ys_{out}\left(s_{out,j},p\right) - ys_{out}\left(s_{out,j'},p\right)\right), \quad \forall j,j' \in J, s_{out,j} \in S_{out,j}, p \in P \tag{6.43}$$

$$ts_{out}\left(s_{out,j},p\right) \leq ts_{out}\left(s_{out,j'},p\right) + H\left(2 - ys_{out}\left(s_{out,j},p\right) - ys_{out}\left(s_{out,j'},p\right)\right), \quad \forall j,j' \in J, s_{out,j} \in S_{out,j}, p \in P \tag{6.44}$$

Similar constraints hold for two water streams entering the storage vessel. Constraints (6.45) ensures that water entering the storage vessel at a later time point corresponds to a later actual time in the time horizon. If two streams are entering the storage vessel at a time point, then the streams must do so at the same absolute time in the time horizon. This is ensured through constraints (6.46) and (6.47)

$$ts_{in}\left(s_{out,j},p\right) \geq ts_{in}\left(s_{out,j'},p'\right) - H\begin{pmatrix} 2 - ys_{in}\left(s_{out,j},p\right) \\ -ys_{in}\left(s_{out,j'},p'\right) \end{pmatrix}, \quad \forall j,j' \in J, s_{out,j} \in S_{out,j}, p,p' \in P, p \geq p' \tag{6.45}$$

$$ts_{in}\left(s_{out,j},p\right) \geq ts_{in}\left(s_{out,j'},p\right) - H\left(2 - ys_{in}\left(s_{out,j},p\right) - ys_{in}\left(s_{out,j'},p\right)\right), \quad \forall j,j' \in J, s_{out,j} \in S_{out,j}, p \in P \tag{6.46}$$

$$ts_{in}\left(s_{out,j},p\right) \leq ts_{in}\left(s_{out,j'},p\right) + H\left(2 - ys_{in}\left(s_{out,j},p\right) - ys_{in}\left(s_{out,j'},p\right)\right), \quad \forall j,j' \in J, s_{out,j} \in S_{out,j}, p \in P \tag{6.47}$$

Constraints (6.48), (6.49) and (6.50) deal with the scheduling of two streams entering and leaving the storage vessel relative to one another. Constraints (6.48) states that if water leaves the storage vessel at a time point after the water entered the storage vessel, the actual time at which this happens must be later in the time horizon. Constraints (6.49) and (6.50) ensure that water entering and leaving the storage vessel at the same time point does so at the same time.

$$ts_{out}\left(s_{out,j},p\right) \geq ts_{in}\left(s_{out,j'},p'\right) - H\left(2 - ys_{out}\left(s_{out,j},p\right) - ys_{in}\left(s_{out,j'},p'\right)\right), \quad \forall j,j' \in J, s_{out,j} \in S_{out,j}, p,p' \in P, p \geq p' \tag{6.48}$$

$$ts_{\text{out}}\left(s_{\text{out},j},p\right) \geq ts_{\text{in}}\left(s_{\text{out},j'},p\right) - H\left(2 - ys_{\text{out}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.49}$$

$$ts_{\text{out}}\left(s_{\text{out},j},p\right) \leq ts_{\text{in}}\left(s_{\text{out},j'},p\right) + H\left(2 - ys_{\text{out}}\left(s_{\text{out},j},p\right) - ys_{\text{in}}\left(s_{\text{out},j'},p\right)\right), \quad \forall j,j' \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.50}$$

The final group of sequencing and scheduling constraints comprise of feasibility constraints and time horizon constraints.

Sequencing Constraints that Associate Production Scheduling and Water Recycle/Reuse

Each water using operation in the time horizon has to be scheduled accordingly, within the overall framework of operation scheduling. This is captured in constraints (6.51), (6.52), (6.53) and (6.54). Constraints (6.51) and (6.52) ensure that unit j is washed immediately after completing a task that produces $s_{\text{out},j}$. Constraints (6.53) is a duration constraints, which defines the starting and ending times of a unit j during the washing operation. Constraints (6.54) stipulates that the washing operation can only commence at time point p if the task producing state $s_{\text{out},j}$ was activated at the previous time point. This could be considered as a complementary constraints to constraints (6.51), (6.52) and (6.53). Constraints (6.55) is optional in the presence of constraints (6.29).

$$tw_{\text{in}}\left(s_{\text{out},j},p\right) \geq t_p\left(s_{\text{out},j},p\right) - H\left(1 - yw\left(s_{\text{out},j},p\right)\right), \quad j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.51}$$

$$tw_{\text{in}}\left(s_{\text{out},j},p\right) \leq t_p\left(s_{\text{out},j},p\right) + H\left(1 - yw\left(s_{\text{out},j},p\right)\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.52}$$

$$tw_{\text{out}}\left(s_{\text{out},j},p\right) = tw_{\text{in}}\left(s_{\text{out},j},p-1\right) + \tau w\left(s_{\text{out},j}\right) yw\left(s_{\text{out},j},p-1\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, s_{\text{in}} \in S_{\text{in}}, p \in P, p > p_1 \tag{6.53}$$

$$yw\left(s_{\text{out},j},p\right) = y\left(s^*_{\text{in},j},p-1\right), \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, s^*_{\text{in},j} \in S^*_{\text{in},j}, s^*_{\text{in},j} \to s_{\text{out},j}, p \in P, p > p_1 \tag{6.54}$$

$$\sum_{s_{\text{out},j}} yw\left(s_{\text{out},j},p\right) \leq 1, \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.55}$$

In the presence of wastewater minimization, the sequence constraints initially presented in Chapter 2 are modified as follows. The following constraints stipulates that the task corresponding to state $s^*_{\text{in},j}$ can only commence once all the previous tasks and their corresponding washing operations are complete.

$$t_{\text{in}}\left(s_{\text{in},j},p\right) \geq \sum_{s_{\text{out},j}} \left\{\left(\tau\left(s^*_{\text{in},j}\right) + \tau w\left(s_{\text{out},j}\right)\right) yw\left(s_{\text{out},j},p'-1\right)\right\},\ \forall j \in J, p, p' \in P,$$
$$p \geq p', p' \geq 2, s^*_{\text{in},j} \in S^*_{\text{in},j}, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j}, s^*_{\text{in},j} \to s_{\text{out},j} \tag{6.56}$$

6.3.3 Objective Function

The objective function used in the mathematical model is either the maximisation of profit or the minimisation of effluent. This is dependent on the nature of the data given for a problem. If the production, e.g. number of batches of each product or total tonnage, is not given then the objective function is the maximisation of profit. However, if this is given then the objective function is the minimisation of effluent.

The constraints given above complete the mathematical formulation for both cases considered. The constraints given above contain a number of nonlinearities, which complicates the solution of resulting models. This was dealt with through the solution procedure discussed below.

6.4 Solution Procedure

One would notice that there are a number of nonlinearities in the above constraints, more specifically in the contaminant mass balances around a unit and the central storage vessel. The nonlinearities arise due to the fact that the outlet concentration of each contaminant may not necessarily be at its respective maximum. Unlike the single contaminant case where one could replace the outlet concentration with the maximum outlet concentration, in the multiple contaminant case the outlet concentration of each contaminant remains a variable. Furthermore, the concentration within the central storage vessel is always variable, since the contaminant mass and mass of water within the vessel changes each time a stream enters or exits the vessel. To deal with this situation the following procedure is considered.

The procedure to find a solution is similar to that proposed by Gouws et al. (2008) and used in Chapter 4. The exact nonlinear model is fully linearised to form a MILP using the relaxation-linearisation technique proposed by Quesada and Grossman (1995). The resulting MILP is solved and its solution is used as an initial solution for the exact MINLP. The MINLP is then solved to find a final solution for the problem. Two possibilities arise from the solution of the exact MINLP. Firstly, the value of the objective function of the exact solution could equal that of the MILP. In this case the solution to the exact model is globally optimal. The second possibility is that the values of the two objective functions are not equal. In this case the solution to the MINLP is only locally optimal. Furthermore, the solution attained from the MILP in this case provides an upper or lower bound, depending on the direction of the optimisation. In this case one might get a better solution using an alternate initial point. However, the question then arises on the generation of such an initial point as to ensure a better solution. The generation of alternate initial starting points fell outside the scope of the work considered. The aim was to find a feasible solution to the MINLP.

6.5 Illustrative Examples

In this section the application of the multiple contaminant methodology is demonstrated though a number of illustrative examples. The first example is solved for both the case where there is no central storage vessel and the case where there is a central storage vessel. The second example is an adapted literature example. Due to the nature of the example it was only solved considering a central storage vessel.

6.5.1 First Illustrative Example

The first illustrative example deals with an operation involving three water using units. There are three contaminants present within the system, with each unit producing wastewater containing each of the three contaminants. The relevant concentration data is given for each unit in Table 6.1. Important to note that each unit produces a unique product, and each product requiring no intermediate material from the other units.

Further data required for the problem is given in Table 6.2. Table 6.2 gives the necessary process durations, maximum amount of water, selling price of each product and raw material costs. The maximum amount of water given in Table 6.2 is calculated using constraints (6.10).

Apart from the above data, further information on the amount of water used in each unit to produce product is as follows. Process 1 requires 3 kg of water to process 1 kg of raw material. Process 2 requires 1 kg of water to process 2 kg of raw material and process 3 requires 1 kg of water to process 1.5 kg of raw material. The

Table 6.1 Concentration data for the first illustrative example

Process	Contaminant	Max. inlet concentration (ppm)	Max. Outlet concentration (ppm)	Mass load(g)
1	1	0	15	675
	2	0	400	18000
	3	0	35	1575
2	1	20	120	3400
	2	300	12500	414800
	3	45	180	4000
3	1	120	220	5600
	2	20	45	1400
	3	200	9500	520800

Table 6.2 Further data for the first illustrative example

Process	Duration (h)	Max water (t)	Selling price (c.u.)	Cost raw (c.u.)
1	2	45	2300	108
2	2.5	34	2000	82
3	1.5	56	1050	95

aforementioned water is the total amount of water entering the unit and is not dependent on the concentration of the water entering the unit. The mass of raw material required is calculated using constraints (6.57), where the amount of water used in a unit is multiplied by the appropriate factor described before. The mass of product produced is the difference in the amount of raw material charged to the unit and the total mass lost to the water in the unit. This is given in constraints (6.58). For this specific example the objective function used is the maximisation of profit, where the profit is the difference in revenue gained from product and the cost of raw materials and the treatment costs of wastewater. The objective is given in constraints (6.59). The wastewater treatment cost is 200 c.u./kg water.

$$m_{\text{in}}\left(s_{\text{in},j},p\right) = mw_{\text{in}}\left(s_{\text{out},j},p\right)\Psi\left(s_{\text{in},j}\right),\quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{in},j} \to s_{\text{out},j}, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.57}$$

$$d\left(s_{\text{out},j},p\right) = m_{\text{in}}\left(s_{\text{in},j},p-1\right) - \sum_c M\left(s_{\text{out},j},c\right) y\left(s_{\text{in},j},p-1\right),\quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{in},j} \to s_{\text{out},j}, s_{\text{out},j} \in S_{\text{out},j}, p \in P, c \in C \tag{6.58}$$

$$R = \sum_p \left(\begin{array}{l} \sum\limits_{s_{\text{out},j}} SP\left(s_{\text{out}}\right) d\left(s_{\text{out},j},p\right) - \sum\limits_{s_{\text{in},j}} CR\left(s_{\text{in}}\right) m_{\text{in}}\left(s_{\text{in},j},p\right) \\ -CE \sum\limits_{s_{\text{out},j}} me\left(s_{\text{out},j},p\right) \end{array} \right),\quad \forall j \in J, s_{\text{out}} \in S_{\text{out}}, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{6.59}$$

This example was solved for both the case where there was no storage available for wastewater and where there was storage available for wastewater.

Solution with No Central Storage Vessel

The example was solved using the solution procedure described in Sect. 4.4. The resulting models were formulated in GAMS 22.0. The MILP was solved using CPLEX 9.1.2 and the DICOPT2 algorithm was used for the exact MINLP. In the DICOPT solution algorithm the MIP solver was CPLEX 9.1.2 and the NLP solver was CONOPT3. The computer used had a Pentium 4, 3.2 GHz processor with 512 MB of RAM. The solution to the problem was found in 2.52 CPU seconds. The resulting formulation had 72 binary variables. The final objective function had a value of 1.860×10^6 c.u., with 552.73 t of effluent being generated. If recycle/reuse of wastewater had not been considered one would have generated 562 t of effluent with the same amount of product. This means that recycling/reuse wastewater allows for a 1.6% savings in the amount of effluent generated.

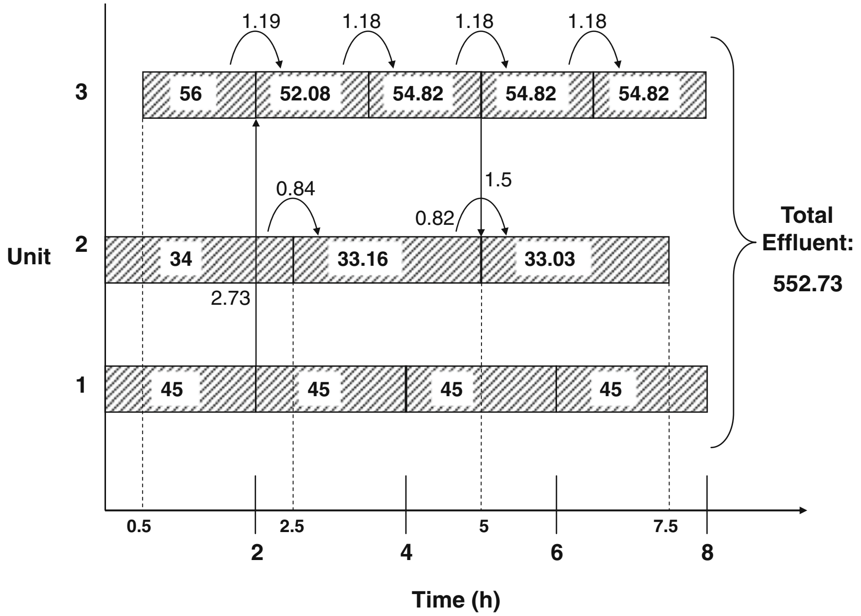


Fig. 6.3 Gantt chart for the first illustrative example with no storage (Majozi and Gouws, 2009)

The resulting schedule for the final solution to the exact MINLP is depicted in the Gantt chart shown in Fig. 6.3. The striped blocks represent the operation of a unit, the bold numbers within each block is the amount of freshwater used and the other numbers represent the amount of water recycled/reused.

One would notice from Fig. 6.3 that units 2 and 3 constantly recycle water. Unit 1 reuses water to unit 3 after 2 h. Further on in the time horizon unit 3 reuses water to unit 2, at time equal to 5 h.

The solution obtained from the exact MINLP is not globally optimal. This is due to the fact that the value of the objective function found in the exact solution is not equal to that of the relaxed MILP. The objective function value in the relaxed solution was 1.8602×10^6 c.u., a slight improvement to that found in the exact model.

Solution with Central Storage

The first illustrative example was solved with the addition of a central storage vessel. The maximum capacity of the storage vessel was 200 t and any of the three units could send or receive water from the central storage vessel.

The true minimum wastewater target is ensured using constraints (6.60), which ensures that at the end of the time horizon there is no water in the storage vessel. Since one does not have any information on events subsequent to the time horizon of interest, it serves no purpose to store any water at the end of the time horizon. Furthermore, water stored at the end of the time horizon gives a false impression of the amount of water saved.

$$q(p) = 0, \quad p = |P| \tag{6.60}$$

As in the previous case the solution procedure described in Sect. 4.4 was used to solve the example. The resulting models were formulated in GAMS 22.0, as with the previous case. CPLEX 9.1.2 was used to solve the MILP and the DICOPT2 solution algorithm was used to solve the exact MINLP. In the DICOPT solution algorithm, CLPEX 9.1.2 was the MIP solver and CONOPT3 the NLP solver. The same processor as the previous example was used to find a solution.

The solution was found in a total time of 522.07 CPU seconds and the resulting formulation had 162 binary variables. The objective function had an optimal value of 1.869×10^6c.u. with 9 time points. The total effluent generated was 504.63 t. As with the previous solution, had recycle/reuse not been considered the total effluent would have been 562 t of water for the same production. This relates to a 10.2% decrease in the amount of effluent generated by recycling/reusing wastewater.

The resulting schedule that achieves the wastewater target is given in Fig. 6.4. Once again, the striped blocks represent each units operation, the bold numbers the amount of freshwater used and the normal case numbers represent the amount of water directly reused. The amount of water sent to storage or used from storage is given by the numbers in italics.

As can be seen in Fig. 6.4, unit 1 reuses water directly to unit 3 at a time equal to 2 h. At the same time unit 1 sends water to the central storage vessel. Unit 1 also sends water to the storage vessel after 4 h in the time horizon. Unit 2 receives water

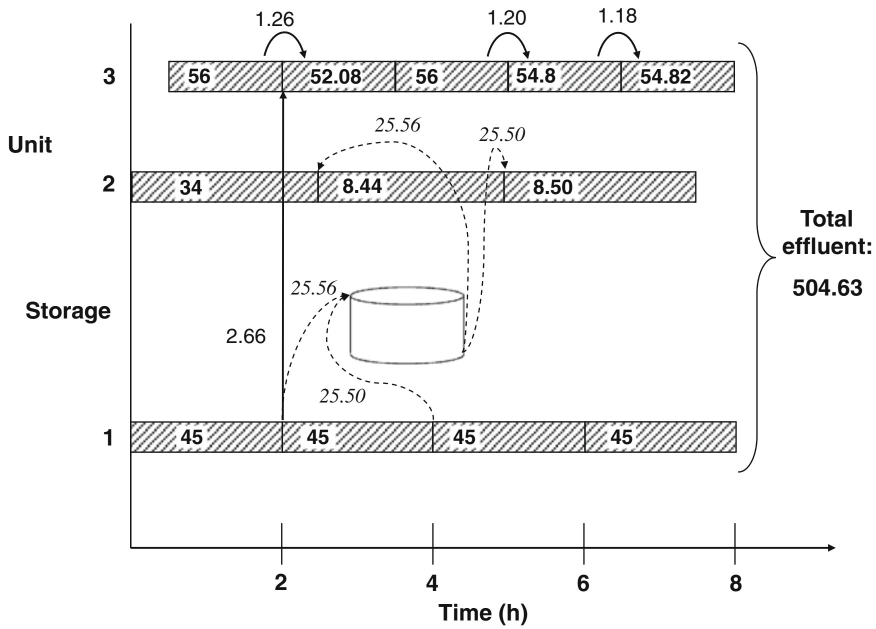


Fig. 6.4 Gantt chart for the first illustrative example with central storage (Majozi and Gouws, 2009)

from the storage vessel at 2.5 h and 5 h. Unit 3 does not make use of the central storage vessel. Unit 3 does recycle water 3 times in the time horizon.

The solution used from the MILP for the starting point was not the optimal solution to the MILP. Instead the solution used had a relative gap of 9.8%, which satisfied default termination criteria in GAMS. When an MILP solution with a zero relative gap was used as the starting point to the MINLP, the resulting solution from the MINLP was worse than that achieved using the initial solution with the higher relative gap. The same holds for solutions with relative gaps lower than 9.8%. Furthermore, the solution time for the MILP with a zero relative gap was excessive. The only explanation for this is that the zero relative gap solution provides an initial starting point in an area of the solution space characterised by a lower local optimum. The high relative gap solution, however, provides an initial point in an area of the solution space characterised by a higher local optimum.

It must also be noted that the solution to the MILP with a zero relative gap was also infeasible, when substituted into the exact MINLP, and the value of the objective function was lower than the previous, it was 1.88×10^6c.u. Therefore, the usage of the higher relative gap solution is allowed. If the solution with a zero relative gap had been feasible, then one would have to use this as the starting point to the exact MINLP model.

As was expected the solution with the central storage vessel, given above, produced less effluent than the solution without the central storage vessel. This is due to the fact that the storage vessel allows for greater reuse opportunities, since inherent time constraints can be bypassed.

In both the above solutions a change in the length of the time horizon will affect the resulting solution. It is very possible that an increase in the length of the time horizon will allow for the production of less effluent, whilst, a decrease in the time horizon could result in an increase in the amount of effluent produced. It is impossible to say if there is a specific trend as the time horizon increases. It is assumed that the time horizon in each problem is fixed during problem specification. The impact of the time horizon on the wastewater reuse opportunities fell outside the scope of the investigation.

The multiple contaminant methodology was applied to a second example. The example and the results are discussed below.

6.5.2 Second Illustrative Example

The second illustrative example is a modified literature example. The example was originally presented by Kim and Smith (2004). The example involves 7 water using operations with three contaminants present in the system. The example was only solved considering a central storage vessel due to the fact that the schedule used by Kim and Smith was retained for this example and there are few direct reuse opportunities within the given schedule. Due to the schedule being known, the objective was to minimise effluent.

In the original example water was constantly used throughout the duration of an operation. This has been changed to suit the model presented beforehand. In the modified example water is produced and used at the ending and beginning of an operation respectively. Furthermore, Kim and Smith (2004) included a second water source, other than freshwater. This was discarded in this example and only freshwater was available to supplement recycle/reused water. Due to these changes and the fact that Kim and Smith (2004) included piping costs and storage costs in their model, the results obtained by Kim and Smith (2004) cannot be readily compared to the results obtained from the application of the derived model.

The required concentration data for the example is given in Table 6.3. Table 6.3 also gives the mass load transferred in each unit and the maximum allowable water in each unit.

The starting and ending times of each operation are given in Table 6.4. One would notice that each operation occurs only once in the 10 h time horizon. The size of the storage vessel available was 2000 t. The example was solved using the solution procedure described previously. The resulting models were formulated in GAMS. The MILP was solved using CPLEX 9.1.2 and the MINLP was solved using the DICOPT2 solution algorithm, with CPLEX 9.1.2 as the MIP solver and CONOPT3 as the NLP solver. The resulting model had 560 binary variables and required a

Table 6.3 Data for the second illustrative example

Operation	Contaminant	Max. inlet conc. (ppm)	Max. outlet conc. (ppm)	Mass load (g)	Max. water (t)
1	C1	0	20	4	200
	C2	0	400	80	
	C3	0	50	10	
2	C1	50	100	15	300
	C2	200	1000	240	
	C3	50	12000	3585	
3	C1	10	200	28.5	150
	C2	50	100	7.5	
	C3	300	1200	135	
4	C1	30	75	9	200
	C2	100	200	20	
	C3	200	1000	160	
5	C1	150	300	15	100
	C2	200	1000	80	
	C3	350	1200	85	
6	C1	0	150	22.5	150
	C2	0	300	45	
	C3	50	2500	367.5	
7	C1	100	200	5	50
	C2	150	1500	67.5	
	C3	220	1000	39	

Table 6.4 Starting and ending times of each process in the second illustrative example

Operation	Start time (h)	End time (h)
1	0	0.5
2	1	2
3	2	3.5
4	1	2
5	4	4.5
6	5.5	6.5
7	8	10

total solution time of 539.9 CPU seconds. The value of the objective function for the exact model was 865.8 t. Had recycle/reuse not been included the amount of effluent would have been 1076.3 t, which relates to a 19.5% reduction in the amount of effluent.

The optimal number of time points used in this example is 8. The objective function value of the MILP, from the first step, was 769.3 t, which is not the same as the exact model. This means that the solution found is only locally optimal.

The schedule including the wastewater reuse is shown in Fig. 6.5. The italic numbers show the amount of water going to storage, the black bold numbers the amount of water coming from storage, the grey bold numbers the amount of freshwater and the normal numbers the amount of water directly reused. One would notice the central storage vessel is used throughout the time horizon by various units.

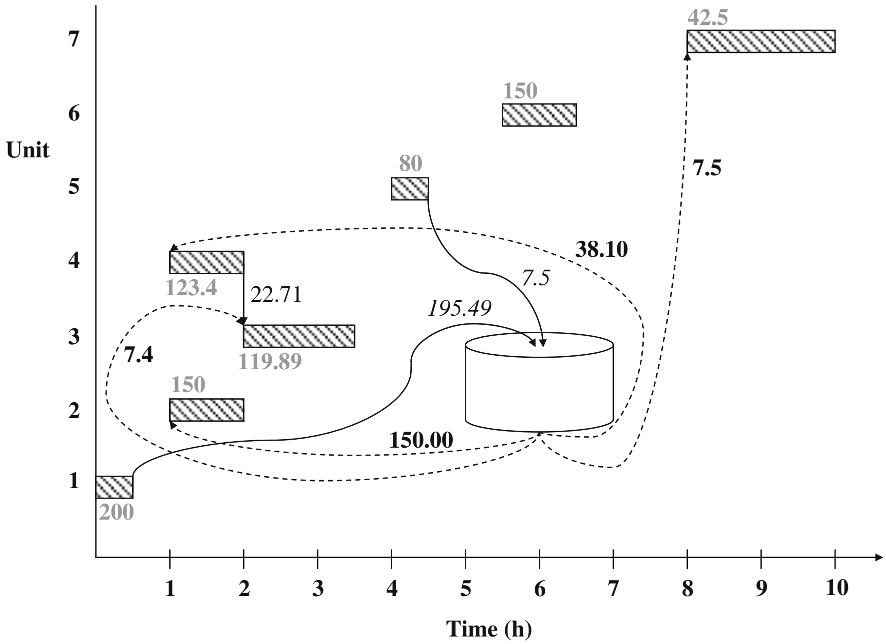


Fig. 6.5 Water reuse for the second illustrative example

6.5.3 Literature Example

The case study involves a well published multipurpose facility which is commonly known as BATCH1 in literature (Kondili et al., 1993). It mainly consists of 3 chemical reactions which take place in 2 common reactors. In addition to the 2 common reactors, the flowsheet also entails the heater and the separator, before and after the reactors, respectively as shown if Fig. 6.6. The STN and SSN for the literature example are given in Figs. 6.7a and b, respectively. The data for this example appears in Table 6.5.

In order to illustrate the capability of the proposed technique, this multipurpose example has been enhanced by including compulsory washing operations after each of the reactions in each of the 2 reactors. The philosophy is that the reactors need to be cleaned after each reaction in order to remove contaminants that are formed as byproducts, so as to ensure product integrity. Data pertaining to cleaning tasks is shown in Table 6.6. The variation in performance in the 2 reactors could be ascribed to differences in design, which is indeed a common encounter in practice. In addition to this data, it is known that freshwater cost is 2 cost units per kg of water whilst the effluent treatment cost is 3 cost units per kg.

The objective function for the literature example is the maximization of a profit function over a 10 h time horizon that takes revenue, freshwater and wastewater treatment costs as follows.

$$\text{Max}\, Z = \sum_{s}\sum_{p} CP(s)\, d(s,p) - CF \sum_{s_{\text{out},j}}\sum_{p} mw_f\left(s_{\text{out},j}, p\right) - CE \sum_{s_{\text{out},j}}\sum_{p} mw_e\left(s_{\text{out},j}, p\right)$$

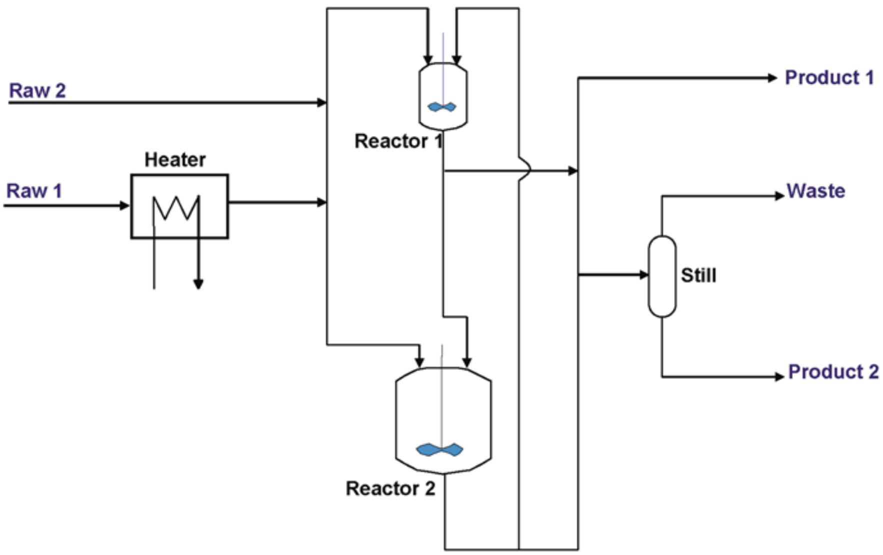


Fig. 6.6 Flowsheet for BATCH1 multipurpose facility

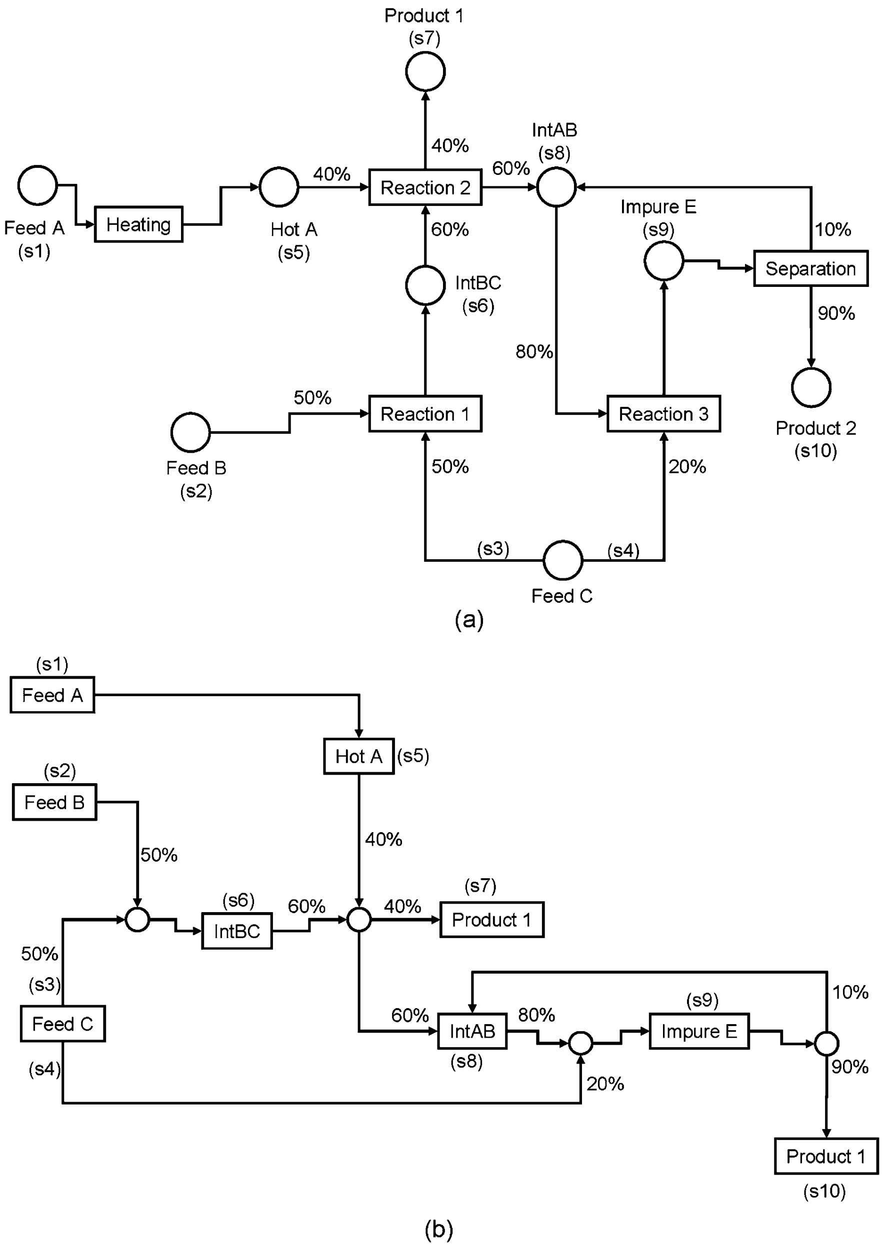


Fig. 6.7 (**a**) STN and (**b**) SSN for the literature example

Table 6.5 Scheduling data for the literature example

Unit	Capacity	Suitability	Mean processing time (h)
Heater ($j = 1$)	100	Heating	1.0
Reactor 1 ($j = 2$)	50	Reaction 1, 2, 3	2.0, 2.0, 1.0
Reactor 2 ($j = 3$)	80	Reaction 1, 2, 3	2.0, 2.0, 1.0
Still ($j = 4$)	200	Separation	1 for product 2, 2 for IntAB
State	Storage capacity	Initial amount	Price
Feed A	Unlimited	Unlimited	0.0
Feed B	Unlimited	Unlimited	0.0
Feed C	Unlimited	Unlimited	0.0
Hot A	100	0.0	0.0
IntAB	200	0.0	0.0
IntBC	150	0.0	0.0
Impure E	200	0.0	0.0
Product 1	Unlimited	0.0	100.0
Product 2	Unlimited	0.0	100.0

Table 6.6 Wastewater minimisation data for the literature example

		Maximum concentration (g contaminant/kg water)		
		Contaminant 1	Contaminant 2	Contaminant 3
Reaction 1	Max. inlet	0.5	0.5	2.3
(Reactor 1)	Max. outlet	1.0	0.9	3.0
Reaction 2	Max. inlet	0.01	0.05	0.3
(Reactor 1)	Max. outlet	0.2	0.1	1.2
Reaction 3	Max. inlet	0.15	0.2	0.35
(Reactor 1)	Max. outlet	0.3	1.0	1.2
Reaction 1	Max. inlet	0.05	0.2	0.05
(Reactor 2)	Max. outlet	0.1	1	12
Reaction 2	Max. inlet	0.03	0.1	0.2
(Reactor 2)	Max. outlet	0.075	0.2	1
Reaction 3	Max. inlet	0.3	0.6	1.5
(Reactor 2)	Max. outlet	2.0	1.5	2.5
		Mass load (g)		
		Contaminant 1	Contaminant 2	Contaminant 3
Reaction 1	Reactor 1	4	80	10
	Reactor 2	15	24	358
Reaction 2	Reactor 1	28.5	7.5	135
	Reactor 2	9	2	16
Reaction 3	Reactor 1	15	80	85
	Reactor 2	22.5	45	36.5
		Duration of washing (h)		
		Reaction 1	Reaction 2	Reaction 3
Reactor 1		0.25	0.50	0.25
Reactor 2		0.30	0.25	0.25

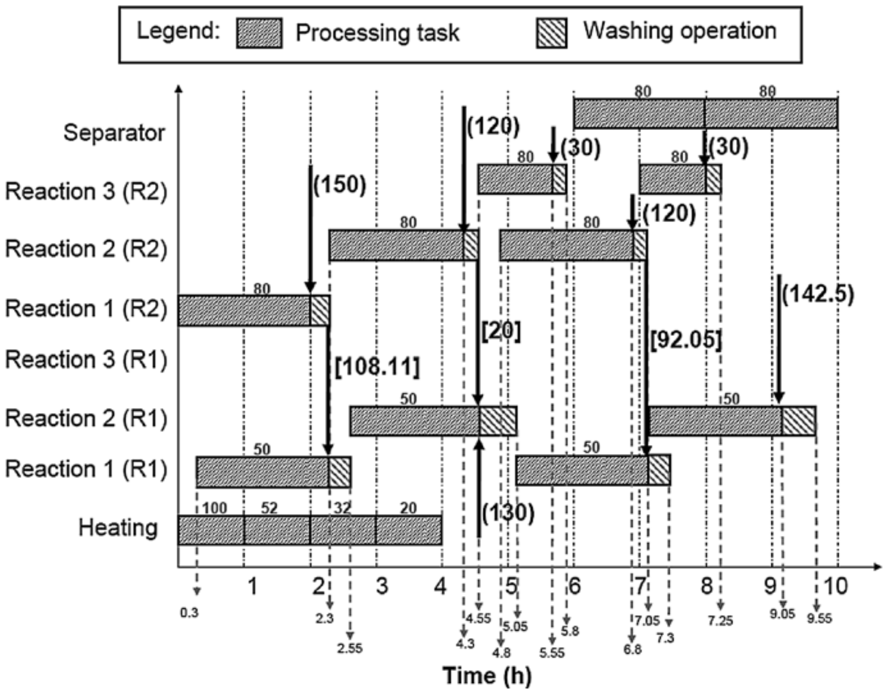


Fig. 6.8 Optimum schedule including recycle and reuse for BATCH1 (Majozi and Gouws, 2009)

Results for the Literature Example

Figure 6.8 shows the optimum production and wastewater minimization schedule for the BATCH1 example. The values in curly brackets represent the amount of freshwater, in kg, used at any given instant. On the other hand, the values in square brackets represent reused water from one operation to another. The values on top of the horizontal blocks represent the amount of material that is processed for production scheduling purposes. This schedule corresponds to an optimum objective value of 21187.5 c.u., which is at most 1.4% from the upper bound set by the relaxed problem. Consequently, this solution is guaranteed to be at most 1.4% from the global optimum. Concomitant with this solution is the usage of 722.5 kg of freshwater. In the absence of possible recycle and reuse, the optimum objective was found to be 20236.1 c.u., which is a *proven global optimum* due to its convergence with the reformulated relaxed objective. This corresponds to 912.8 kg of freshwater use. Therefore, in addition to improving the economic objective function, exploitation of recycle/reuse opportunities results in almost 21% reduction in freshwater use and effluent generation.

6.6 Conclusions

The methodology presented in this chapter deals with wastewater minimisation where the wastewater contains multiple contaminants. The methodology determines the minimum wastewater target and the corresponding schedule that achieves the wastewater target. The methodology is derived for two cases, presented in two models. The first case deals with wastewater minimisation without a central storage vessel and the second deals with wastewater minimisation where there is a central storage vessel.

Due to the fact that the outlet concentration of each contaminant is not at its maximum, the models derived each take the form of an MINLP. The nonlinearities are linearised using the relaxation-linearisation technique proposed by Quesada and Grossman (1995). The linearised model that results is used to generate an initial solution for the exact non-linear model.

Two illustrative examples were considered. The first example was solved for both the case where there was a central storage vessel and not. The reduction in wastewater where there was no central storage vessel was only 1.6%. With the inclusion of a central storage vessel this increased to 10%. The second example stems from a literature example presented by Kim and Smith (2004). The schedule was given in the example and the example solved only considering a central storage vessel. The amount of water saved was 19.5%. In the more complex BATCH1 literature example, this methodology resulted in 20.8% savings in wastewater generation, whilst taking scheduling considerations into account. It is worthy of note that one cannot compare water savings in a single contaminant process to water savings where there are multiple contaminants, even though they share similar characteristics. The amount of wastewater that can be saved will be highly dependant on the maximum inlet concentrations of the individual components.

The main drawback of the methodology is that the resulting models could reach large proportions due to the direct dependency of the size of the resulting model on the number of time points used. The larger the number of time points, the larger the problem. In situations where the number of time points is large, the resulting model could reach such proportions that a solution cannot be found.

6.7 Exercise

Task: Determine the minimum amount of freshwater used and wastewater generated for the following real-life industrial scale case study.
Problem description: Industrial case study

The facility considered in the industrial case study is a pharmaceuticals production plant, which produces a wide variety of consumer products, e.g. shampoos and creams, and female sanitary products. The plant was chosen due to the fact that production was carried out in batch mode. In a year the plant uses on average 90000 m^3 of water, with approximately 60–70% of this discarded as effluent. An on-site treatment facility pre-treats wastewater produced from the site to a level where the water can be discharged into the municipal water system.

The plant is divided into three main areas. The first is the toiletries mixing area, the second the pharmaceuticals mixing area and the third the female sanitary products area. The production of the female sanitary products required no water during the production phase, the area was therefore ignored in the case study.

The toiletries mixing area, the main focus of the study, comprises of 5 mixers producing liquid products and 1 mixer producing solid products in the form of powders. The mixer producing the powders was ignored in the case study due to the fact that there was no water used in any production steps. Of the 5 mixers producing liquid products, one mixer is dedicated to the mixing of oil products and used no water in the production steps. It was also ignored in the study. The remaining 4 mixers could produce anyone of 49 products. For simplicity these products were grouped into four main groups, namely, shampoos, creams, lotions and deodorants. Products in each group have very similar physical characteristics and in many cases, the products share the same base composition, the variations arise in the colorants and specialised ingredients added. Certain products are preferentially mixed in certain mixing vessels. This is due to the fact that the mixers are not identical and certain mixer characteristics suit the production of certain products. The products produced in the toiletries mixing area are produced in fixed batch sizes of 2 t.

The general production procedure in the toiletries area is as follows. Raw material is charged to a mixer. The raw material is then mixed until the required physical characteristics are obtained. Once a product is mixed it is removed and stored in either a dedicated storage vessel or a disposable storage container. The mixers are then cleaned.

There are 10 dedicated storage vessels in the area and numerous disposable storage vessels. Each product is stored for a minimum of 12 h before it is packaged to allow for settling of possible solids and to allow for the product to cool.

Water is used in the area in product, for heating and cooling and to clean the mixers and storage vessels. In this study the only water considered is the water used for cleaning. This due to the fact that the only opportunity for wastewater reuse occurs in cleaning operations.

The final area that uses water is the pharmaceuticals area. The pharmaceuticals area can be divided into three sections. The first section produces petroleum jelly based products, the second sanitising products and the third specialised creams and lotions. The first section comprises of two mixers and water is only used for heating or cooling. The second section has 3 mixers and water usage occurs mainly in product. The final section is comprised of a single mixer, where water is used for cleaning, in product and heating and cooling. In the first two sections water reuse is impossible due to the fact that there are no operations that can reuse water. Due to the fact that the third section is only comprised of a single mixer and it is undesirable to mix wastewater from the pharmaceuticals section with wastewater from the toiletries section, the mixer was not included in the case study.

Apart from the mixing areas there are numerous packaging lines. Due to the relatively small amount of water these use, they were not included into the case study.

A full breakdown of water usage at the plant is given in Fig. 6.9. The case study focussed on the reuse of water produced from cleaning operations in the toiletries mixing area.

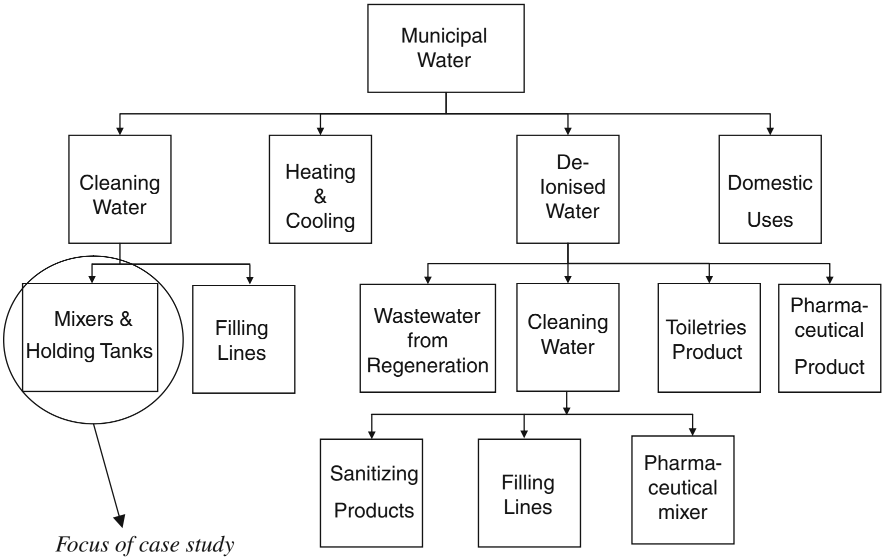


Fig. 6.9 Water usage at the pharmaceutical plant

6.7.1 A Typical Cleaning Procedure in the Toiletries Mixing Area

It is important to fully understand the physical cleaning cycle and the implications this has on the application of the methodologies. The same cleaning cycle is followed for both the mixing vessels and the intermediate storage vessels.

The cleaning procedure employed at the industrial site is a four step procedure with a total duration of 31 min. The first step is a rinse with pure water. This step removes the bulk of the product residue left in a mixer or storage vessel and consequently the concentration of contaminants in the water is high. Once this step is complete a chemical wash step is then started. In this step the vessel is cleaned using a water and surfactant solution. This wash removes any residue remaining after the first rinse. The temperature of the water in this step is elevated to facilitate the cleaning. The third step is a clean water rinse, as with the first step. This removes any water and surfactant solution still present in the mixer from the previous step. The final step is a sanitising step. In this step the vessel is rinsed with water and sanitising agent solution at elevated temperature to ensure the required sanitisation is achieved.

Practically wastewater reuse in another cleaning cycle can only be done in steps two, three and four. The water from the first step has a very high contaminant concentration, and reuse of this water in another cleaning operation would not achieve significant wastewater reductions. The wastewater from the other three steps has relatively low concentrations of contaminants and reuse of the water is feasible. In the application of the derived methodologies to the case study, the last three steps were modelled as one processing step.

The industrial case study was centred on the four mixers found in the toiletries area, as described above. Due to the multiple products found in the area, the residues within the system were not all the same (Table 6.7). This coupled with compatibility considerations between the various contaminants in the wastewater meant that the multiple contaminant wastewater minimisation methodology was suited to the problem.

The maximum outlet concentrations from each mixer are given in Table 6.8. The maximum outlet concentrations given in the table stem from approximate process data. It is important to notice that there is maximum outlet concentration defined for only one component from each mixer. This is since each mixer will at any point only contain residue of a specific product. The inlet and outlet concentrations of the other products will be equal. The outlet concentration from each mixer varies due to the varying nature of the products.

Due to the varying nature of the products, one has some compatibility considerations on which the type of wastewater can be reused for a certain residue. This is the case with the wastewater produced from mixer 2 containing the deodorant residue. The deodorant is by nature hydrophobic, and therefore comes easily out of solution. The reuse of wastewater containing deodorant in a mixer with any other residue is undesirable because it is very possible that the deodorant residue in the wastewater can come out of solution when the wastewater is mixed with the residue of the other product. The deodorant residue will then be left in the mixer and the mixer will have to be re-cleaned. However, wastewater containing the deodorant can be reused in a cleaning operation in mixer 2, containing another deodorant residue. This compatibility is controlled by setting the maximum inlet concentrations to appropriate levels. In this case the maximum inlet concentration of the deodorant in mixers 1, 3 and 4 is set to zero.

The compatibility is not the only consideration when determining the maximum inlet concentrations for a cleaning operation. In the case where the residue in the wastewater and the residue in the mixer are compatible one still has a maximum contaminant load that can enter with the water for a cleaning operation. This is due to the fact that one still has to obey certain mass transfer constraints.

The maximum inlet concentration of each mixer is given in Table 6.9. As can be seen the inlet concentration for the deodorant is zero for mixers 1, 3 and 4. This is due to the incompatibility of this residue with the other residues as discussed previously. Since wastewater reuse was never attempted at the facility under question, the maximum inlet concentrations were approximated. These values are conservative and still allow for the required cleanliness.

Table 6.7 Residue mass left in each mixer in the industrial case study

Mixer Number	Residue Mass (kg)
1 (Shampoos)	15
2 (Deodorants)	15
3 (Lotions)	30
4 (Creams)	70

Table 6.8 Maximum outlet concentrations of the wastewater from a cleaning operation

Contaminant and mixer	Maximum outlet concentration (kg product/kg water)
Shampoos (1)	0.04
Deodorants (2)	0.045
Lotions (3)	0.05
Creams (4)	0.06

Table 6.9 Maximum inlet concentrations for a cleaning operation in the industrial case study

	Shampoo (kg product/kg water)	Deodorant (kg product/kg water)	Lotion (kg product/kg water)	Creams (kg product/kg water)
Mixer 1	0.014	0	0.007	0.0035
Mixer 2	0.014	0.0035	0.007	0.007
Mixer 3	0.014	0	0.007	0.0035
Mixer 4	0.014	0	0.007	0.0035

In a typical multiple contaminant problem the maximum amount of water that can be used, while still obeying any concentration constraints, is determined by a limiting component and there is generally contaminant mass added for each contaminant present. In this problem, contaminant mass is only added to the water for one contaminant, namely the residue left from the specific product in a mixer. This then makes the limiting component in each mixer the component that leaves residue in the mixer. For example mixer 1 has shampoo as the limiting contaminant, since this is the only component which leaves a residue in the mixer. The maximum amount of water for each mixer is given for each mixer in Table 6.10.

The typical production requirement over a 24 h time period is given in Table 6.11. Also given in this table are average production times for each product. The duration of a washout was taken as 30 min. A 10 t central water storage vessel was available for wastewater storage.

Table 6.10 Maximum water used for a washout in the industrial case study

Mixer	Maximum water (kg)
1	576.9
2	361.4
3	1238.9
4	697.6

Table 6.11 Number of batches to be produced in the industrial case study

Product	Number of batches	Duration (h)
Shampoo	2	7
Deodorant	3	5.5
Lotion	1	11
Cream	2	11

References

Gouws, J.F., Majozi, T., Gadalla, M., 2008. Flexible mass transfer model for water minimization in batch plants. Chem. Eng. Process., 47: 2323–2335

Kim, J.K., Smith, R., 2004. Automated design of discontinuous water systems. Process Safety Environ. Protect., 82: 238–248

Kondili, E., Pantelides, C. C., Sargent, R. W. H., 1993. A general algorithm for short-term scheduling of batch operations. I. MILP formulation. Comput. Chem. Eng., 17(2): 211–227

Majozi, T., 2005. Wastewater minimization using central reusable storage in batch plants. Comput. Chem. Eng. 29: 1631–1646

Majozi, T., Gouws, J., 2009. A mathematical optimization approach for wastewater minimization in multiple contaminant batch plant. Comput. Chem. Eng. doi:10.1016/j.compchemeng.2009.06.008

Majozi, T., Zhu, X. 2001. A novel continuous-time MILP formulation for multipurpose batch plants.1. Short-term scheduling. Ind. Eng. Chem. Res. 40(23): 5935–5949

Quesada, I., Grossmann, I.E. 1995. Global optimization of bilinear process networks with multicomponent flows. Comp. Chem. Eng. 19: 1219–1242

Chapter 7
Wastewater Minimisation Using Multiple Storage Vessels

Overview In Chapter 6 the most generalised model for wastewater minimisation in batch plants was presented. The model is capable of simultaneously generating an optimum production schedule and a water recycle/reuse network that is concomitant with minimum wastewater generation in multipurpose batch plants. However, the mathematical formulation presented in Chapter 6 is only applicable in the presence of only 1 reusable water storage facility. The implication, therefore, is that there exist no incompatibility issues among the present contaminants. If there is interaction between or among the existing contaminants, a need for multiple storage units might prove necessary. The methodology presented in this chapter focuses on the wastewater minimisation problem where there are multiple contaminants present in a system with multiple storage options. These storage options arise from the nature of the wastewater present and the type of contaminants present in the wastewater. The first section of this chapter provides background to the methodology developed later in the chapter. In the second section the formal problem statement is given. The mathematical formulation is presented in the third section. The fourth section presents a number of illustrative examples to which the methodology was applied. The fifth section presents conclusions that can be drawn from the chapter. Worthy of mention is that the mathematical formulation as presented in this chapter is directly applicable to multiproduct rather than multipurpose batch plants.

7.1 Multiple Storage Vessel Background

Multiple contaminant wastewater minimisation was dealt with in Chapter 6 wherein it was assumed that there were no compatibility issues with wastewater produced containing different contaminants and that all the contaminants present in the system could be mixed. However, in certain instances this is not the case. Such instances arise when there are different contaminants in each stream that cannot necessarily be mixed, or there is a unit that can only receive wastewater contaminated with a certain contaminant. In such instances it would be advantageous to have

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_7,

multiple storage vessels dedicated to the storage of different types of wastewater. To fully understand this, one has to consider the types of processing units found in industry.

Generally processing units can be divided into two groups based on the type of wastewater produced. The first group are those processing units that produce wastewater with a single, but different, contaminant and the second group are those processing units that generate wastewater with more than one contaminant present. A processing facility can comprise of a mixture of units from the above groups or comprise of units from a single group. Depending on the type of units within the operation one can use different storage options to increase water reuse. Storage vessels can be dedicated to the storage of wastewater with a specific contaminant present or wastewater contaminated with multiple contaminants present.

The model is derived to take into consideration the possibility of multiple storage vessels which are dedicated to the storage of certain wastewater. The formulation shares some of the characteristics of the multiple contaminant model presented in the previous chapter. This is due to the fact that both formulations have roots in the scheduling methodology derived by Majozi and Zhu (2001). Furthermore, the uneven discretization of the time horizon is used as the time representation.

7.2 Problem Statement

The problem addressed in the formulation can be stated as follows:
Given:

i) the contaminant mass load of each contaminant,
ii) the necessary cost and stoichiometric data,
iii) the maximum inlet and outlet concentrations of each contaminant to each unit,
iv) the number of storage vessels and their maximum inlet concentration of each contaminant into the vessel,
v) the available units and their capacities,
vi) the time horizon of interest, and
vii) the maximum capacity of storage available for water reuse,

determine the production schedule that achieves the minimum amount of wastewater generation through recycle/reuse and use of the available storage opportunities.

7.3 Mathematical Formulation

The mathematical formulation presented in this chapter involves the following sets, variables and parameters.

Sets

$P = \{p|p = \text{time point}\}$
$J = \{j|j = \text{unit}\}$
$C = \{c|c = \text{contaminant}\}$
$U = \{u|u = \text{reusable water storage vessel}\}$
$S_{\text{in}} = \{S_{\text{in}}|S_{\text{in}} = \text{input state into any unit}\}$
$S_{\text{out}} = \{S_{\text{out}}|S_{\text{out}} = \text{output state from any unit}\}$
$S_{\text{in},j} = \left\{S_{\text{in},j}|S_{\text{in},j} = \text{input state into unit } j\right\} \subseteq S_{\text{in}}$
$S_{\text{out},j} = \left\{S_{\text{out},j}|S_{\text{out},j} = \text{output state from unit } j\right\} \subseteq S_{\text{out}}$
$S = \{s|s \, \text{isastate}\} = S_{\text{in},j} \cup S_{\text{out},j}$

Continuous Variables

$cc_{\text{in}}\,(j,c,p)$	inlet concentration of contaminant c of unit j at time point p
$cc_{\text{out}}\,(j,c,p)$	outlet concentration of contaminant c of unit j at time point p
$css_{\text{in}}\,(u,c,p)$	inlet concentration of contaminant c into storage vessel u at time point p
$css_{\text{out}}\,(u,c,p)$	outlet concentration of contaminant c from storage vessel u at time point p
$f_e\,(j,p)$	mass of effluent into unit j at time point p
$f_f\,(j,p)$	mass of freshwater to unit j at time point p
$f_p\,(j,p)$	mass water produced at time point p from unit j
$f_r\,\left(j',j,p\right)$	mass of water recycled to unit j from j' at time point p
$f_u\,(j,p)$	mass of water into unit j at time point p
$fs_{\text{in}}\,(j,u,p)$	mass of water to storage vessel u from unit j at time point p
$fs_{\text{out}}\,(u,j,p)$	mass of water from storage vessel u to unit j at time point p
$\text{mraw}\,(s_{\text{in}},j,p)$	mass of raw material used in unit j at time point p
$\text{mprod}\,(s_{\text{out}},j,p)$	mass of product produced from unit j at time point p
$t_p\,(s_{\text{out}},j,p)$	time at which unit j finishes operating at time point p
$t_{rr}\,\left(j',j,p\right)$	time at which water is recycled from unit j to unit j' at time point p
$t_u\,(s_{\text{in}},j,p)$	time at which unit j starts operating at time point p
$tss_{\text{in}}\,(j,u,p)$	time at which water goes to storage vessel u from unit j at time point p
$tss_{\text{out}}\,(u,j,p)$	time at which water leaves storage vessel u to unit j at time point p

Binary Variables

$y\,(s_{\text{in}}, j, p)$	binary variable for usage of unit j at time point p
$y_{rr}\left(j', j, p\right)$	binary variable for usage of recycle from unit j to unit j' at time point p
$yss_{\text{in}}\,(j, u, p)$	binary variable for usage of water into storage vessel u from unit j at time point p
$yss_{\text{out}}\,(u, j, p)$	binary variable for usage of water from storage vessel u to unit j at time point p

Parameters

CE	effluent treatment cost (cost /kg water)
$CC_{\text{in}}^{\text{max}}\,(c, j)$	maximum inlet concentration of contaminant c in unit j
$CC_{\text{out}}^{\text{max}}\,(c, j)$	maximum outlet concentration of contaminant c from unit j
$CR\,(s_{\text{in}})$	cost of raw material (c.u./kg raw material)
$Cs_{\text{in}}^{\text{max}}\,(c,u)$	maximum inlet concentration of contaminant c into storage vessel u
$CS_{\text{out}}^{0}\,(c,u)$	initial concentration of contaminant c in the storage vessel u
$F_{\text{w}}^{\text{max}}\,(j)$	maximum amount of water to unit j
H	time horizon of interest
$M_{\text{lost}}\,(c, j)$	mass load of contaminant c in unit j
$Qm^{\text{O}}\,(u)$	initial amount of water stored in a storage vessel u
$Q^{\text{max}}\,(u)$	maximum storage capacity of a water storage vessel u
$SP\,(s_{\text{out}})$	selling price of product (c.u./kg product)
$\tau\,(j)$	mean processing time of unit j

The mathematical formulation comprises of a number of mass balances and scheduling constraints. Due to the nature of the processes involved, the time aspect is prevalent in all the constraints in some form or another. A superstructure is used in the derivation of the mathematical model, as discussed in the following section. A description of the sets, variables and parameters can be found in the nomenclature list.

7.3.1 Superstructure for the Multiple Storage Vessel Methodology

The mathematical methodology is based on the superstructure given in Fig. 7.1. This superstructure is similar to that used for the multiple contaminant model with a central storage vessel, however, in this case there are multiple storage vessels available for wastewater storage. The effect of this can be seen in Fig. 7.1a, which depicts any water using operation. The water going to a unit is the sum of freshwater, water from direct recycle/reuse and water from the various storage vessels. This is similar

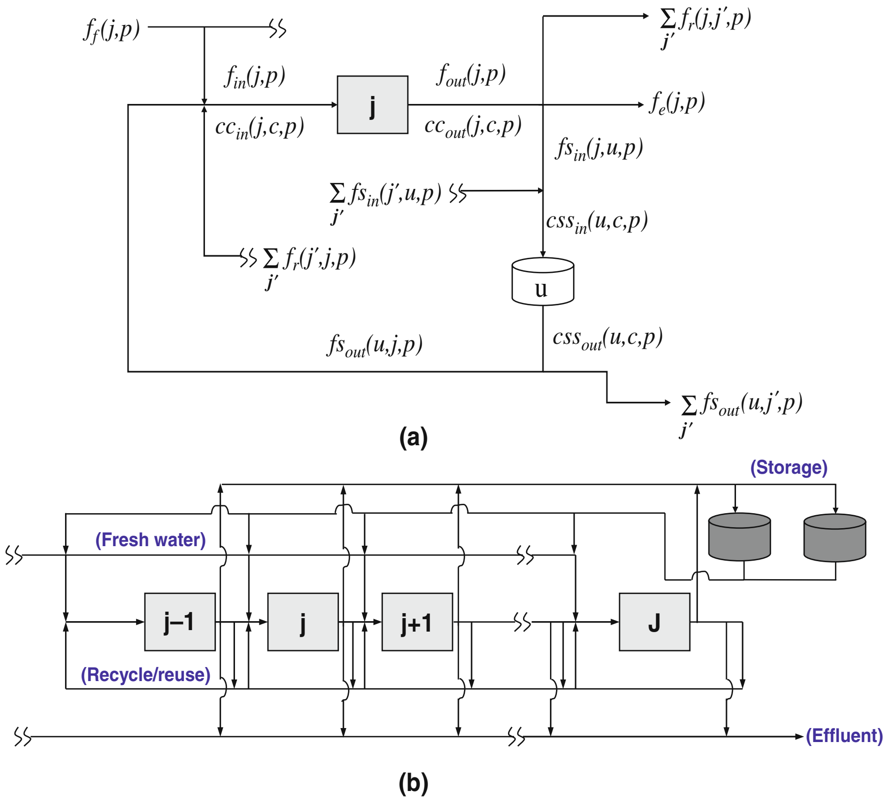


Fig. 7.1 Superstructure for multiple storage vessel formulation (Gouws and Majozi, 2008)

for water leaving the unit. Figure 7.1b depicts the overall plant superstructure. What is apparent from the overall plant superstructure is the presence of multiple storage vessels.

As with the multiple contaminant wastewater minimisation model, the mathematical model for multiple storage vessels comprises of two modules, namely, a mass balance module and a scheduling module. The constraints that comprise the mass balance module are described first.

7.3.2 Mass Balance Constraints

The mass balances that are first considered are those that deal with the mass flow around a unit. The first of these is an inlet water balance, given in constraint (7.1). The water into a unit is the sum of the directly recycled/reused water, freshwater and water from the various storage vessels. The outlet water balance is presented in constraint (7.2). This constraint states that water leaving a processing unit is either directly recycled/reused, discarded as effluent or sent to one or more storage vessels for later reuse. In this formulation it is assumed that each operation does not generate or consume water, thus, the inlet water is equal to the outlet water amount. The possibility of water generation and consumption can easily be accounted for by the inclusion of a water balance over the inlet and outlet of a unit.

$$f_u(j,p) = \sum_{j'} f_r(j',j,p) + f_f(j,p) + \sum_u fs_{\text{out}}(u,j,p) \quad \forall j,j' \in J, p \in P, u \in U \tag{7.1}$$

$$f_p(j,p) = \sum_{j'} f_r(j,j',p) + f_e(j,p) + \sum_u fs_{\text{in}}(j,u,p) \quad \forall j,j' \in J, p \in P, u \in U \tag{7.2}$$

Due to the presence of contaminants in the system, contaminant balances also have to be derived around a unit. The first of these is an inlet contaminant balance, given in constraint (7.3). This constraint states that the contaminant mass entering a unit comprises of the contaminant mass present in the directly recycled/reused water and the contaminant mass present in water from various storage vessels. Important to note is that constraint (7.3) holds for every contaminant in the system.

$$cc_{\text{in}}(j,c,p) f_u(j,p) = \sum_{j'} f_r(j',j,p)\, cc_{\text{out}}(j',c,p) + \sum_u fs_{\text{out}}(u,j,p) css_{\text{out}}(u,c,p) \quad \forall j,j' \in J, p \in P, c \in C, u \in U \tag{7.3}$$

A contaminant balance over a unit is given in constraint (7.4). This constraints states that the mass of contaminant c leaving a unit is the sum of the mass of contaminant c that entered the unit and the mass load of contaminant c added to the water due to the operation in the unit.

$$f_p(j,p)\, cc_{\text{out}}(j,c,p) = f_u(j,p-1)\, cc_{\text{in}}(j,c,p-1) + M_{\text{lost}}(c,j) y\left(s_{\text{in},j}, p-1\right), \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P, p > p_1, c \in C \tag{7.4}$$

In each unit there is a maximum amount of water that can be used while still removing the required mass load and obeying the respective maximum inlet and outlet concentrations. The maximum amount of water based on concentrations for a unit is determined using constraint (7.5). Due to the presence of multiple contaminants in the system, the maximum amount of water used in a unit will be determined by the limiting contaminant. The limiting contaminant is that contaminant that requires the highest quantity of water compared to that required by the other contaminants. The outlet concentration of the limiting contaminant, at the optimal solution, will generally be at its maximum, while the other contaminants will not.

$$F_w^{\max}(j) = \max_{c \in C}\left(\frac{M_{\text{lost}}(j,c)}{CC_{\text{out}}^{\max}(j,c) - CC_{\text{in}}^{\max}(j,c)}\right), \forall j \in J \tag{7.5}$$

Every source of water going into a unit has to be limited to the maximum amount determined in the previous constraint. Constraint (7.6) ensures that the total water into a unit obeys the maximum allowable. Whilst constraints (7.7) and (7.8) ensure that water reused to a unit, directly and indirectly, obeys the maximum amount allowable. In the case where the water requirement for each process is fixed, constraint (7.6) changes, and the current operator is replaced with an equal to sign.

$$f_u\left(j,p\right) \leq F_w^{\max}\left(j\right) y\left(s_{\text{in},j},p\right) \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{7.6}$$

$$f_r\left(j',j,p\right) \leq F_w^{\max}\left(j\right) y_{rr}\left(j',j,p\right) \quad \forall j',j \in J, p \in P \tag{7.7}$$

$$fs_{\text{out}}\left(u,j,p\right) \leq F_w^{\max}\left(j\right) yss_{\text{out}}\left(u,j,p\right) \quad \forall j \in J, p \in P, u \in U \tag{7.8}$$

The inlet and outlet concentration of each contaminant c around a unit is restricted to its respective maxima through constraints (7.9) and (7.10).

$$cc_{\text{in}}\left(j,c,p\right) \leq CC_{\text{in}}^{\max}\left(j,c\right) y\left(s_{\text{in},j},p\right) \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P, c \in C \tag{7.9}$$

$$\begin{gathered} cc_{\text{out}}\left(j,c,p\right) \leq CC_{\text{out}}^{\max}\left(j,c\right) y\left(s_{\text{in},j},p-1\right) \\ \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P, p > p_1, c \in C \end{gathered} \tag{7.10}$$

Apart from the mass balances associated with water, one also has to consider a product mass balance. Constraint (7.11) states that the amount of product leaving a unit is the amount of raw material that entered the unit less the total contaminant mass load transferred to the water stream.

$$\begin{gathered} \text{mprod}\left(s_{\text{out},j},p\right) = \text{mraw}\left(s_{\text{in}},j,p-1\right) - \left(\sum_c M_{\text{lost}}\left(j,c\right)\right) y\left(s_{\text{in},j},p-1\right), \\ \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P, p > p_1, c \in C \end{gathered} \tag{7.11}$$

Constraints dealing with water to and from each storage vessel also have to be considered. Constraint (7.12) is a water balance over a storage vessel. This constraint states that the amount of water stored in a storage vessel at a time point comprises of water stored in the vessel from the previous time point and the difference between the amount of water entering and leaving the vessel. The water balance around the storage vessel for the first time point is given in constraint (7.13).

$$\begin{gathered} qm\left(u,p\right) = qm\left(u,p-1\right) + \sum_j fs_{\text{in}}\left(j,u,p\right) - \sum_j fs_{\text{out}}\left(u,j,p\right), \\ \forall j \in J, p \in P, p > p_1, u \in U \end{gathered} \tag{7.12}$$

$$qm\left(u,p\right) = Qm^O\left(u\right) - \sum_j fs_{\text{out}}\left(u,j,p\right), \forall j \in J, p \in P, p = p_1, u \in U \tag{7.13}$$

The inlet concentration of a contaminant c into a storage vessel is defined in constraint (7.14). This constraint states that the inlet concentration of contaminant c is the ratio of the contaminant mass in water sent to a vessel to the total amount of water entering a vessel. Constraint (7.15) is included to control the type of contaminants that enter a vessel and the maximum allowable inlet concentration into a storage vessel. By selecting the appropriate maximum inlet concentration one can dedicate certain vessels to the storage of water containing certain contaminants. A contaminant balance over a storage vessel is given in constraint (7.16). This constraint states that the mass of contaminant c stored in a vessel at a time point is the amount stored from the previous time point and the difference in the amount entering and leaving the storage vessel. The initial concentration of contaminant c in a storage vessel is given in constraint (7.17).

$$css_{\text{in}}(u,c,p) = \frac{\sum_j \left(fs_{\text{in}}(j,u,p)\, cc_{\text{out}}(j,c,p)\right)}{\sum_j fs_{\text{in}}(j,u,p)}, \quad \forall\, j,j' \in J, p \in P, c \in C, u \in U \tag{7.14}$$

$$css_{\text{in}}(u,c,p) \le Cs_{\text{in}}^{\max}(u,c) \quad \forall\, p \in P, c \in C, u \in U \tag{7.15}$$

$$\begin{aligned} css_{\text{out}}(u,c,p)\, qm(u,p) = {} & css_{\text{out}}(u,c,p-1)\, qm(u,p-1) \\ & + \sum_j fs_{\text{in}}(j,u,p)\, css_{\text{in}}(u,c,p) \\ & - \sum_j fs_{\text{out}}(u,j,p)\, css_{\text{out}}(u,c,p), \\ \forall\, j \in J, p \in P, p > p_1, c \in C, u \in U & \end{aligned} \tag{7.16}$$

$$css_{\text{out}}(u,c,p) = CS_{\text{out}}^{O}(u,c) \quad \forall\, u \in U, c \in C, p \in P, p = p_1 \tag{7.17}$$

Constraint (7.18) ensures that the amount of water stored in a storage vessel at any time point is less than the capacity of a storage vessel. Constraint (7.19) ensures that the amount of water entering a storage vessel is less than the capacity of the storage vessel. At the end of the time horizon each storage vessel should not be storing any water, since it is unknown when the water will be used. This is ensured through constraint (7.20). Furthermore, any stored water at the end of the time horizon will give a false impression of the amount of water saved.

$$qm(u,p) \le Q^{\max}(u) \quad \forall\, p \in P, u \in U \tag{7.18}$$

$$fs_{\text{in}}(j,u,p) \le Q^{\max}(u)\, yss_{\text{in}}(j,u,p) \quad \forall\, j \in J, p \in P, u \in U \tag{7.19}$$

$$qm(u,p) = 0 \quad \forall\, p \in |P|, u \in U \tag{7.20}$$

The constraints presented above for the mass balances around a storage vessel hold for each storage vessel within the system.

One would notice that there are a number of nonlinear terms in the above constraints, specifically in the contaminant balance constraints. The linearisation technique used to remove these nonlinearities is that proposed by Quesada and Grossman (1995), the general form of this linearization technique can be found in Appendix A. During the application of the model to the illustrative examples, presented in Section 7.4, it was noticed that only one term required linearisation for a solution to be found. This term being the $f_p\,(j,p)\,cc_{\text{out}}\,(j,c,p)$ term in constraint (7.4). Due to this, the solution technique proposed for the multiple contaminant methodology was not used.

The constraints presented above complete the constraints necessary for the mass balances. Due to the discontinuous nature of the operation under consideration, constraints also have to be derived that capture the essence of time. These are presented in the scheduling module below.

7.3.3 Scheduling Constraints

The constraints that comprise the scheduling module of the model are divided into four groups, namely, task scheduling, direct recycle/reuse scheduling, storage scheduling and time horizon constraints.

Task Scheduling Constraints

The task scheduling constraints used in this model are similar to those that have been presented in full in Chapter 2. However, the reader should note that the time variables in these constraints relate to water using operations only. These constraints will therefore not be discussed here, since the full description of the constraints can be found in the preceding chapters, particularly Chapter 2. However, these constraints are presented below.

$$t_u\left(s_{\text{in},j},p\right) \geq t_p\left(s_{\text{out},j},p\right) - H\left(2 - y\left(s_{\text{in},j},p\right) - y\left(s_{\text{in},j}{}',p-1\right)\right),\quad \forall j \in J, s_{\text{in},j}{}', s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j}, p \in P, p > p_1 \tag{7.21}$$

$$t_p\left(s_{\text{out},j},p\right) = t_u\left(s_{\text{in},j},p-1\right) + \tau\left(s_{\text{in}}\right) y\left(s_{\text{in},j},p-1\right),\quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j}, s_{\text{in}} \in S_{\text{in}}, p \in P, p > p_1 \tag{7.22}$$

$$t_u\left(s_{\text{in},j},p\right) \geq t_u\left(s_{\text{in},j}{}',p'\right) - H\left(2 - y\left(s_{\text{in},j},p\right) - y\left(s_{\text{in},j}{}',p'\right)\right),\quad \forall j \in J, s_{\text{in},j}, s_{\text{in},j}{}' \in S_{\text{in},j}, p, p' \in P, p \geq p' \tag{7.23}$$

$$t_p\left(s_{\text{out},j},p\right) \geq t_p\left(s_{\text{out},j},p'\right) - H\left(2 - y\left(s_{\text{in},j},p\right) - y\left(s_{\text{in},j}{}',p'\right)\right),\quad \forall j \in J, s_{\text{in},j}, s_{\text{in},j}{}' \in S_{\text{in},j}, s_{\text{out},j}, s_{\text{out},j}{}' \in S_{\text{out},j}, p, p' \in P, p \geq p' \tag{7.24}$$

Recycle/Reuse Scheduling

The constraints used for the scheduling of the recycle/reuse in the model are similar to those used in the previous multiple contaminant wastewater minimisation methodology, (see constraints (6.31), (6.32), (6.33), (6.34) and (6.35)). Again, the reader is alerted to the fact that the following constraints apply to water using operations, i.e. they are not as generalised as in Chapter 6. This is due to the fact that direct recycle/reuse is treated in the same manner as previously. A description of the constraints for recycle/reuse will, therefore, not be given here. However, for ease of reference these are presented below.

$$y_{rr}\left(j,j',p\right) \le \sum_{s_{\text{in}}} y\left(s_{\text{in},j}',p\right),\ \ \forall j,j' \in J, s_{\text{in},j} \in S_{\text{in},j} p \in P \tag{7.25}$$

$$\begin{aligned} &t_{rr}\left(j,j',p\right) \le t_p\left(s_{\text{out},j},p\right) + H\left(2 - y_{rr}\left(j,j',p\right) - y\left(s_{\text{in},j},p-1\right)\right), \\ &\forall\, j,j' \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j}, p \in P, p > p_1 \end{aligned} \tag{7.26}$$

$$\begin{aligned} &t_{rr}\left(j,j',p\right) \ge t_p\left(s_{\text{out},j},p\right) - H\left(2 - y_{rr}\left(j,j',p\right) - y\left(s_{\text{in},j},p-1\right)\right), \\ &\forall\, j,j' \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j}, p \in P, p > p_1 \end{aligned} \tag{7.27}$$

$$\begin{aligned} &t_{rr}\left(j',j,p\right) \le t_u\left(s_{\text{in},j},p\right) + H\left(2 - y_{rr}\left(j',j,p\right) - y\left(s_{\text{in},j},p\right)\right), \\ &\forall j,j' \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \end{aligned} \tag{7.28}$$

$$\begin{aligned} &t_{rr}\left(j',j,p\right) \ge t_u\left(s_{\text{in},j},p\right) - H\left(2 - y_{rr}\left(j',j,p\right) - y\left(s_{\text{in},j},p\right)\right), \\ &\forall j,j' \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \end{aligned} \tag{7.29}$$

Storage Scheduling Constraints

The first constraints considered in the storage scheduling deal with water going from a unit to a storage vessel. Constraints (7.30) and (7.31) ensure that the time at which water is transferred from a unit to a storage vessel coincides with the time at which the water is produced from the unit, i.e. the finishing time of the unit. Furthermore, water can only be transferred from a unit to a storage vessel at a time point provided the unit started operating at the previous time point. This is ensured through constraint (7.32). A unit can, however, operate without sending water to a storage vessel. This possibility is included in constraint (7.32).

$$\begin{aligned} &tss_{\text{in}}\left(j,u,p\right) \ge t_p\left(s_{\text{out}},j,p\right) - H\left(2 - yss_{\text{in}}\left(j,u,p\right) - y\left(s_{\text{in}},j,p-1\right)\right), \\ &\forall\, j \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{out}} \in S_{\text{out}}, p \in P, p > p_1, u \in U \end{aligned} \tag{7.30}$$

$$\begin{aligned} &tss_{\text{in}}\left(j,u,p\right) \le t_p\left(s_{\text{out}},j,p\right) + H\left(2 - yss_{\text{in}}\left(j,u,p\right) - y\left(s_{\text{in}},j,p-1\right)\right), \\ &\forall\, j \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{out}} \in S_{\text{out}}, p \in P, p > p_1, u \in U \end{aligned} \tag{7.31}$$

$$\begin{aligned} &yss_{\text{in}}\left(j,u,p\right) \le \sum_{s_{\text{in}}} y\left(s_{\text{in}},j,p-1\right), \\ &\forall\ s_{\text{in}} \in S_{\text{in}}, j \in J, p \in P, p > p_1, u \in U \end{aligned} \tag{7.32}$$

Constraints (7.33) and (7.34) are similar to constraints (7.30) and (7.31), however, they deal with water transferred from a storage vessel to a unit. Constraints (7.33) and (7.34) ensure that the time at which water is transferred from a storage vessel to a unit coincides with the time at which the receiving unit starts operating, i.e., the starting time of a unit. Constraint (7.35) ensures that if water is transferred from a storage vessel to a unit, the unit is operating at that time point. However, a unit can operate without receiving water from storage.

$$tss_{\text{out}}(u,j,p) \geq t_u(s_{\text{in}},j,p) - H\left(2 - yss_{\text{out}}(u,j,p) - y(s_{\text{in}},j,p)\right),$$
$$\forall\, j \in J, s_{\text{in}} \in S_{\text{in}}, u \in U, p \in P \tag{7.33}$$

$$tss_{\text{out}}(u,j,p) \leq t_u(s_{\text{in}},j,p) + H\left(2 - yss_{\text{out}}(u,j,p) - y(s_{\text{in}},j,p)\right),$$
$$\forall\, j \in J, s_{\text{in}} \in S_{\text{in}}, u \in U, p \in P \tag{7.34}$$

$$yss_{\text{out}}(u,j,p) \leq \sum_{s_{\text{in}}} y(s_{\text{in}},j,p), \forall\, s_{\text{in}} \in S_{\text{in}}, j \in J, p \in P \tag{7.35}$$

Scheduling constraints have to be derived to account for the timing of multiple streams leaving a storage vessel. Constraint (7.36) ensures that water leaving a storage vessel at a later time point does so at a later absolute time in the time horizon. Constraints (7.37) and (7.38) ensure that the time at which two streams leave a storage vessel at a time point corresponds to the same time for each.

$$tss_{\text{out}}(u,j,p) \geq tss_{\text{out}}(u,j',p') - H\left(2 - yss_{\text{out}}(u,j,p) - yss_{\text{out}}(u,j',p')\right),$$
$$\forall\, j,j' \in J, u \in U, p, p' \in P, p > p' \tag{7.36}$$

$$tss_{\text{out}}(u,j,p) \geq tss_{\text{out}}(u,j',p) - H\left(2 - yss_{\text{out}}(u,j,p) - yss_{\text{out}}(u,j',p)\right),$$
$$\forall\, j,j' \in J, u \in U, p \in P \tag{7.37}$$

$$tss_{\text{out}}(u,j,p) \leq tss_{\text{out}}(u,j',p) + H\left(2 - yss_{\text{out}}(u,j,p) - yss_{\text{out}}(u,j',p)\right),$$
$$\forall\, j,j' \in J, u \in U, p \in P \tag{7.38}$$

Constraints (7.39), (7.40) and (7.41) deal with the scheduling of multiple streams entering a storage vessel. Constraint (7.39) ensures that water entering a storage vessel at a later time point does so at a later absolute time in the time horizon. Furthermore, the time at which two streams enter a storage vessel at the same time point must correspond to the same absolute time in the time horizon. This is ensured through constraints (7.40) and (7.41).

$$tss_{\text{in}}(j,u,p) \geq tss_{\text{in}}(j',u,p') - H\left(2 - yss_{\text{in}}(j,u,p) - yss_{\text{in}}(j',u,p')\right),$$
$$\forall\, j,j' \in J, u \in U, p, p' \in P, p > p' \tag{7.39}$$

$$tss_{\text{in}}(j,u,p) \geq tss_{\text{in}}(j',u,p) - H\left(2 - yss_{\text{in}}(j,u,p) - yss_{\text{in}}(j',u,p)\right),$$
$$\forall\, j,j' \in J, u \in U, p \in P \tag{7.40}$$

$$tss_{\text{in}}(j,u,p) \leq tss_{\text{in}}(j',u,p) + H\left(2 - yss_{\text{in}}(j,u,p) - yss_{\text{in}}(j',u,p)\right),$$
$$\forall\, j,j' \in J, u \in U, p \in P \tag{7.41}$$

The final constraints considered in the scheduling of the storage vessels deal with the relation between streams entering and leaving a storage vessel. The first of these, given in constraint (7.42), ensures that water leaving a storage vessel at a later time point to when water entered the vessel does so at a later actual time in the time horizon. Constraints (7.43) and (7.44) ensure that the time at which water enters and leaves a storage vessel at a time point corresponds to the same time.

$$tss_{\text{out}}(u,j,p) \geq tss_{\text{in}}(j',u,p') - H\left(2 - yss_{\text{out}}(u,j,p) - yss_{\text{in}}(j',u,p')\right),$$
$$\forall\, j,j' \in J, u \in U, p, p' \in P, p > p' \tag{7.42}$$

$$tss_{\text{out}}(u,j,p) \geq tss_{\text{in}}\left(j',u,p\right) - H\left(2 - yss_{\text{out}}(u,j,p) - yss_{\text{in}}\left(j',u,p\right)\right), \quad \forall j,j' \in J, u \in U, p \in P \tag{7.43}$$

$$tss_{\text{out}}(u,j,p) \leq tss_{\text{in}}\left(j',u,p\right) + H\left(2 - yss_{\text{out}}(u,j,p) - yss_{\text{in}}\left(j',u,p\right)\right), \quad \forall j,j' \in J, u \in U, p \in P \tag{7.44}$$

The scheduling constraints presented above hold for each storage vessel in the process. It must be noted that there is no interaction between the various storage vessels, since each storage vessel is independent of the other.

Time Horizon Constraints

The final constraints necessary to complete the scheduling model are constraints that ensure all operations occur within the time horizon. These are given in constraints (7.45), (7.46), (7.47), (7.48) and (7.49).

$$tss_{\text{in}}(j,u,p) \leq H, \quad \forall j \in J, u \in U, p \in P \tag{7.45}$$

$$tss_{\text{out}}(u,j,p) \leq H, \quad \forall j \in J, u \in U, p \in P \tag{7.46}$$

$$t_u\left(s_{\text{in},j},p\right) \leq H, \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{7.47}$$

$$t_p\left(s_{\text{out},j},p\right) \leq H, \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{7.48}$$

$$t_{rr}\left(j,j',p\right) \leq H, \quad \forall j,j' \in J, p \in P \tag{7.49}$$

The constraints given above complete the scheduling module of the model. The final part of the model that needs consideration is the objective function.

7.3.4 Objective Function

The objective function used is dependent on the production information provided. If the production requirements are given for the time horizon of interest, e.g. the tonnage required, required number of batches, etc., then the objective function is the minimisation of effluent. If the production is not given then the objective is the maximisation of profit, where profit is defined as the difference between revenue from product, cost of raw material and effluent treatment costs. A typical profit objective function is given in constraint (7.50). This objective function is used in the illustrative examples.

$$\max R = \sum_p \left(\sum_{s_{\text{in}}, s_{\text{out}}, j} SP(s_{\text{out}})\, m_p\left(s_{\text{out},j},p\right) - \sum_j CR(s_{\text{in}})\, m_u\left(s_{\text{in},j},p\right) - CE \sum_j f_e(j,p) \right)$$
$$\forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{7.50}$$

The constraints described above complete the mathematical formulation for wastewater minimisation using multiple storage vessels. The application of the formulation to various illustrative examples is described below.

7.4 Illustrative Examples for the Multiple Storage Vessel Model

Three illustrative examples are presented to demonstrate the application and effectiveness of the proposed multiple storage vessel methodology. Each example deals with a different case of multiple contaminant problem, which will become apparent in the description of each.

7.4.1 First Illustrative Example

The first illustrative example deals with the minimisation of wastewater in an operation that involves three processing units, with each unit producing a distinct product. Wastewater produced from units 1 and 2 each contain single, but different, contaminants. Wastewater from unit 1 contains contaminant *C1* and wastewater from unit 2 contains contaminant *C2*. Unit 3 produces wastewater that contains three contaminants, namely, contaminants *C1*, *C2* and *C3*. Unit 1 can only receive water contaminated with contaminant *C1*. Similarly, unit 2 can only receive water contaminated with contaminant *C2*.

Two wastewater storage vessels are considered in the problem, with one storage vessel dedicated to the storage of wastewater containing contaminant *C1* only and the other dedicated to the storage of wastewater containing contaminant *C2* only. A third storage vessel for the storage of wastewater containing contaminants *C1*, *C2* and *C3* is not considered, due to a process requirement that no wastewater containing contaminant *C3* be stored. It is important to note that during the problem formulation the usage of each specific storage vessel by the various units is unknown. Furthermore, in this example, multiple dedicated storage vessels have been included to ensure the most opportunities for wastewater reuse.

The concentration data required for the problem is given in Table 7.1. Table 7.1 gives the maximum inlet and outlet concentration for each unit and the mass load in each unit.

Table 7.2 provides the relevant cost data for product and raw material, as well as the average duration of each process and the maximum allowable water for each process calculated using constraint (7.5). The effluent treatment cost is 200 c.u. per ton.

The water into each storage vessel is controlled by setting the appropriate maximum inlet concentration of each contaminant into a storage vessel to an appropriate level. For the first storage vessel, the maximum inlet concentration of contaminant *C1* is set to 500 000 ppm, while the maximum inlet concentration of contaminants *C2* and *C3* is set to 0 ppm. For the second storage vessel the maximum inlet

Table 7.1 Concentration data for the first illustrative example

Unit	Contaminant	Max. inlet conc. (ppm)	Max. outlet conc. (ppm)	Mass load(g)
1	1	5	15	675
	2	0	0	0
	3	0	0	0
2	1	0	0	0
	2	50	100	25000
	3	0	0	0
3	1	120	220	5600
	2	200	450	14000
	3	200	9500	520800

Table 7.2 Cost data for the first illustrative example

Unit	Cost of product (c.u.)	Cost of raw material (c.u.)	Duration (h)	Maximum water (t)
1	2300	108	2	67.5
2	2000	82	2.5	50
3	1050	95	1.5	56

concentration of contaminant *C2* is set to 500 000 ppm and the maximum inlet concentration of contaminants *C1* and *C3* is set to 0 ppm. Each storage vessel has a capacity of 200 t.

A further specification given was the amount of water required by each process to process raw material. In unit 1, 1 kg of water was used to process 3 kg of raw material. In unit 2, 1 kg of water was required to process 2 kg of raw material and in unit 3, 1 kg of water was required to process 1.5 kg of raw material. The time horizon of interest was 8 h.

The example was formulated in GAMS 22.0 and solved using the DICOPT2 solution algorithm, with CPLEX 9.1.2 as the MIP solver and CONOPT3 as the NLP solver. The model was solved using a Pentium 4 3.2 GHz processor and required 16.8 CPU seconds to find a solution. DICOPT did 4 major iterations to find the final solution. The optimal number of time points was 8, which resulted in 192 binary variables for the model.

The resulting schedule is shown in Fig. 7.2. The bold numbers in Fig. 7.2 show the amount of freshwater used by each process in tons, while the italic numbers the amount of water directly reused and the normal numbers the amount of water to and from each storage vessel.

The schedule shown in Fig. 7.2 produces 461.7 t of effluent and which results in an objective function value of 2.65×10^6 c.u. If wastewater recycle/reuse had not been considered the resulting effluent would have been 34% more, for the same amount of product. The solution given above cannot be seen as a globally optimal solution since the model is an MINLP. One would notice in Fig. 7.2 that storage vessel one only receives water from unit 1 and storage vessel two only receives

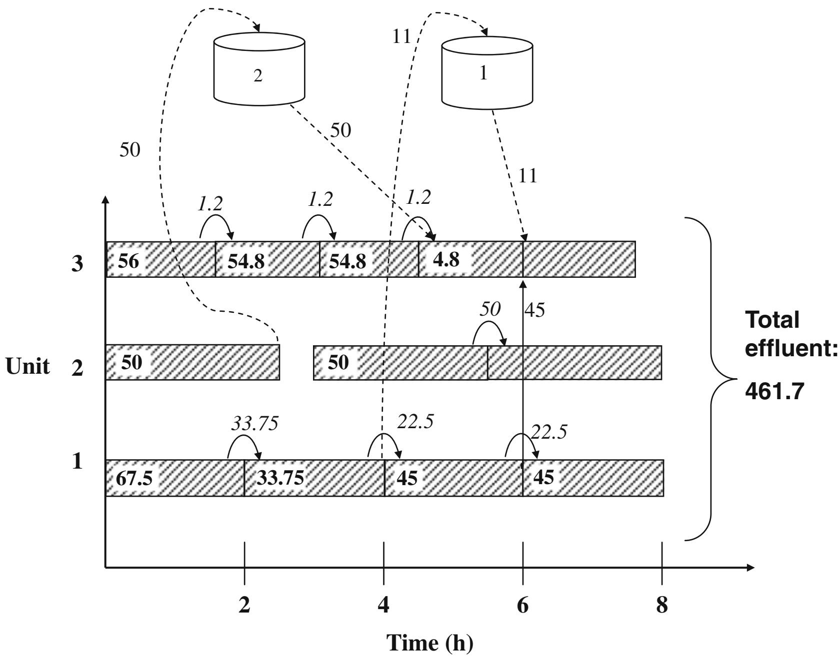


Fig. 7.2 Resulting schedule for the first example using multiple storage vessels (Gouws and Majozi, 2008)

water from unit 2. Furthermore, no water from unit 3 is stored. The water stored in both storage vessel one and two is only used by unit 3. It is important to note that a change in the schedule might result in different usage of the dedicated wastewater storage to that given in Fig. 7.2. Therefore, one cannot remove a storage vessel or change its dedication based on the solution given above.

The effect of increasing the time horizon length was not investigated in this example, or any of the examples that follow. The results are, therefore, unique in each scenario. The cyclic operation of each process was also not investigated.

7.4.2 Second Illustrative Example

The second illustrative example deals with wastewater minimisation in an operation involving four processes. Wastewater produced from processes 1 and 2 contain the same single contaminant *C1*, while wastewater produced from processes 3 and 4 contain multiple contaminants, namely, contaminants *C1*, *C2* and *C3*. Processes 1 and 2 cannot receive water contaminated with multiple contaminants, while processes 3 and 4 can receive any type of wastewater. Table 7.3 provides the required maximum inlet and outlet concentrations for each process as well as the mass load of each contaminant in each process.

Table 7.3 Concentration and durations for the second illustrative example

Unit	Contaminant	Max. inlet conc. (ppm)	Max. outlet conc. (ppm)	Mass load (g)
1	1	5	15	675
	2	0	0	0
	3	0	0	0
2	1	50	100	25000
	2	0	0	0
	3	0	0	0
3	1	120	220	5600
	2	200	450	14000
	3	200	9500	520800
4	1	500	900	32
	2	50	150	5
	3	100	200	6

As with the previous example, two storage vessels are considered in the example. The first storage vessel can only store wastewater containing contaminant *C1* and the second can store wastewater containing any contaminants. The maximum inlet concentration into the first storage vessel was set to 500 000 ppm for contaminant *C1* and zero for contaminants *C2* and *C3*. The maximum inlet concentration for the second storage vessel was set to 10 000 ppm for contaminants *C1*, *C2* and *C3*, since this vessel could store wastewater contaminated with any contaminants. The required cost data is given in Table 7.4, as well as the duration of each process and the maximum amount of water that can be used by each unit. The effluent treatment costs were taken as 200 c.u. per ton of effluent.

As with the previous example the amount of water used to process raw material is given. The ratios used for the first three units, units 1, 2 and 3, are exactly the same as in the previous example, namely, 3 kg raw material requires 1 kg of water in unit 1, 2 kg raw material requires 1 kg of water in unit 2 and 1.5 kg raw material requires 1 kg of water in unit 3. For unit 4, 2.25 kg raw material requires 1 kg of water. The time horizon of interest is 8 h.

The model for the second illustrative example was formulated in GAMS 22.0 and solved using the DICOPT2 solution algorithm. The MIP solver used was CPLEX

Table 7.4 Cost data for the second multiple storage vessel example

	Cost of product (c.u.)	Cost of raw material (c.u.)	Duration (h)	Maximum water (t)
Unit 1	2300	108	2	67.5
Unit 2	2000	82	1.3	50
Unit 3	1050	95	3	56
Unit 4	1300	102	2.33	80

9.1.2 and the NLP solver used was CONOPT3. The resulting model had 324 binary variables and the optimum number of time points was 9. The solution was found in 297.06 CPU seconds using a Pentium 4 3.2 GHz processor. The value of the objective function was 3.63×10^6 c.u. Three major iterations were required by the DICOPT algorithm to find a final solution. The resulting amount of effluent was 495.79 t. Had recycle/reuse of wastewater not been considered the amount of water used would have been 922 t, for the same amount of product. This relates to a wastewater saving of 46%. The resulting schedule is given in Fig. 7.3. Once again, the bold numbers represent the amount of freshwater used, the italic numbers the amount of water directly recycled/reused and the normal numbers the amount of water reused through storage.

One would notice from Fig. 7.3 that neither units 1 or 2 receive wastewater from storage, but each rather sends water to both storage vessels one and two. Unit 3 does not send any water to storage, but only receives water from storage. From the schedule it is apparent that a dedicated vessel storing only water containing contaminant *C1* is not necessary due to the fact that the units that can only receive wastewater contaminated with contaminant *C1* do not receive water from storage. This is not, however, apparent at the start of the problem. It is impossible to determine whether a dedicated storage vessel is required or not at the onset of a problem.

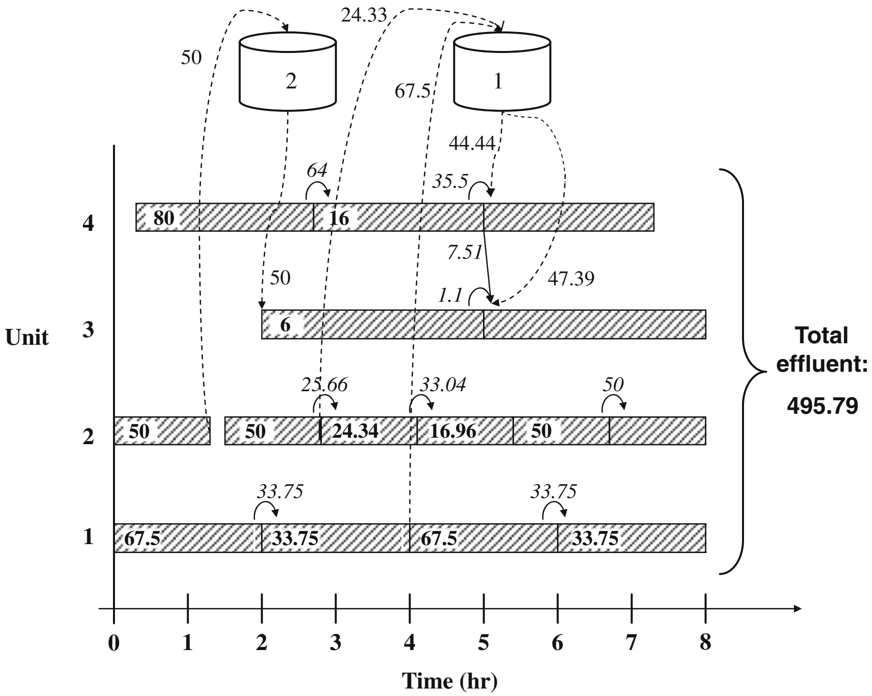


Fig. 7.3 Schedule for the second multiple storage vessel illustrative example

7.4.3 Third Illustrative Example

A final illustrative example involves the scheduling of four operations within a plant. The operations in units 1 and 4 produce wastewater containing contaminants *C1* only and operations in units 2 and 3 produce wastewater containing contaminant *C2* only. Two storage vessels are considered in this example, with storage vessel one dedicated to the storage of wastewater containing contaminant *C1* and storage vessel two dedicated to storing wastewater containing contaminant *C2*. This was ensured by setting the maximum inlet concentration of contaminant *C1* to 500 000 ppm and contaminant *C2* to 0 ppm in storage vessel one, and the reverse for storage vessel two. Table 7.5 gives the required concentration data for each unit, maximum water and duration of each operation.

The cost data used for this example is the same as that given in Table 7.4 from the previous example and the effluent treatment cost was once again 200 c.u. per ton of water. The time horizon for this example is 9 h. The water and raw material relations used in this example were the same as those used in the previous example.

The resulting model was formulated in GAMS 22.0 and solved using the DICOPT2 solution algorithm, with CPLEX 9.1.2 as the MIP solver and CONOPT3 as the NLP solver. Due to the fact that there was only one contaminant present in any water stream, the outlet concentration of the contaminants in each wastewater stream was set to the maximum for each contaminant. The solution to the problem was found in 279.4 CPU seconds using the same processor as previous. The resulting model had 288 binary variables with the optimum number of time points being 8. The value of the objective function was 2.965×10^6 c.u., which relates to 569.67 t of effluent. Had recycle/reuse not been considered, for the same amount of product, the amount of effluent would have been 871.11 t. This means that the amount of effluent generated was reduced by 34%. The resulting schedule is shown in Fig. 7.4. As with the previous figures, the bold numbers represent the amount of freshwater used in each unit, the italic numbers the amount of water directly recycled/reused and the normal numbers the amount of water reused through storage.

Table 7.5 Concentration data for the third multiple storage vessel example

Unit	Max outlet concentration (ppm)		Max inlet concentration (ppm)		Mass load (g)		Max water (t)	Process duration (h)
	C1	**C2**	**C1**	**C2**	**C1**	**C2**		
1	10	0	0	0	675	0	67.5	2
2	0	100	0	50	0	2500	50	2.5
3	0	450	0	200	0	14 000	56	1.5
4	200	0	100	0	7600	0	72	3

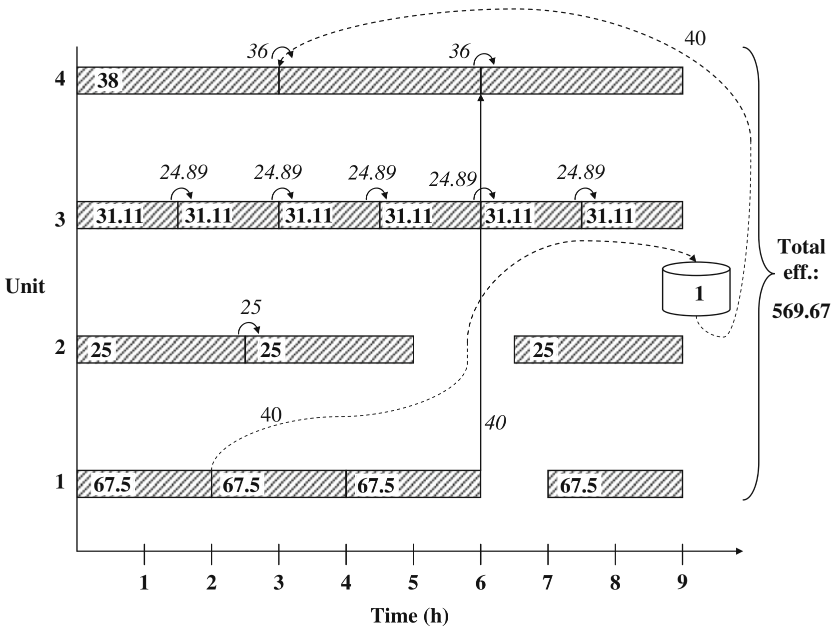


Fig. 7.4 Resulting schedule for third illustrative example

One would notice that storage vessel two is not required in the schedule. This is due to the fact that unit 3 recycles water to itself throughout the time horizon and unit 2 recycles water once in the time horizon. Water is reused through storage vessel one only once in the time horizon.

7.5 Conclusions

The formulation presented in this chapter provides a means to take different storage considerations into account. The methodology is capable of dealing with multiple contaminants and determines the minimum wastewater target in such a system as well as the corresponding schedule.

The methodology takes the form of an MINLP, which must be linearised to find a solution. The linearization method used was the relaxation-linearization technique proposed by Quesada and Grossman (1995). During the application of the formulation to the illustrative examples it was found that only one term required linearization for a solution to be found.

The application of the formulation to the illustrative examples showed that the formulation allows for remarked reduction in the amount of effluent generated. In the first example a reduction of 34% is achieved, while a reduction of 46% is achieved in the second example. A reduction of 34% is achievable in the schedule proposed in the third example.

The main problem with the formulation is that the formulation could reach large proportions for larger problems. This is due to the way in which the binary variables are defined and the fact that the size of the resulting model is directly dependent on the number of time points. For a large number of processing units and storage vessels the number of binary variables is large, which relates to longer solution times. Furthermore, as the number of time points increase, so does the size of the resulting model. For a large number of time points the problem could become intractable.

7.6 Exercise

Task: Revisit the first example and perform the same analysis over 10 and 12 h time horizons. Note that the time points will have to be increased with the longer time horizon. Assess the computational intensity of the mathematical formulation as a function of the time horizon.

References

Gouws, J., Majozi, T., 2008. Impact of multiple storage in wastewater minimisation for multi-contaminant batch plants: towards zero effluent. Ind. Eng. Chem. Res., 47: 369–379

Majozi, T., Zhu, X., 2001. A novel continuous-time MILP formulation for multipurpose batch plants.1. Short-term scheduling. Ind. Eng. Chem. Res. 40(23): 5935–5949

Quesada, I., Grossmann, I. E., 1995. Global optimization of bilinear process networks with multicomponent flows. Comput. Chem. Eng. 19: 1219–1242

Chapter 8
Zero Effluent Methodologies

Overview Current environmental regulations advocate minimal adverse impact from industrial emissions, including effluent. Ideally, industrial operations should not produce any effluent so as to avoid environmental impact. The mathematical formulations presented in the foregoing chapters of this textbook are aimed at reduction, but not elimination of effluent. The mathematical formulation presented in this chapter deals with a special form of wastewater reuse which has the potential to produce near zero effluent. This special form of wastewater reuse pertains to situations where water constitutes a significant part of the final product as encountered in pharmaceutical and agrochemical facilities. The chapter is structured as follows. The first section provides some background to the operation considered. Section 8.2 describes the superstructure used for the mathematical formulations. The zero effluent scheduling formulation is given in Section 8.3 and the zero effluent synthesis formulation is given in Section 8.4. Section 8.5 presents illustrative examples which demonstrate the application of the derived formulations. Conclusions are given in the final section.

8.1 Zero Effluent Operation

Generally, wastewater is produced at the end of a batch and then reused for the processing of a subsequent batch of material. In general, the unit operation considered in wastewater minimisation in batch processes both consumes and produces water. Furthermore, the operations considered are generally mass transfer type processes, where mass is transferred to the water stream due to the operation occurring in a unit. In such operations wastewater reuse between the various units is governed by timing considerations and inlet and outlet concentration limitations. However, in processes where water consumption and wastewater production do not occur in the same operation, a unique opportunity arises in that the wastewater could be reused in the operations that consume water.

The type of operation in which such an opportunity arises occurs often where product is produced, with water as a large constituent, and wastewater is generated

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_8,

from cleaning operations. Wastewater from a cleaning operation contains residue of the product that was produced in the unit previously. The wastewater generated can be reused in the same product as the residue in the wastewater, provided the wastewater does not compromise product integrity.

Reuse of this nature has a number of advantages. Firstly, wastewater produced is reduced since it is reused in product. In situations where the amount of water used in product is more than the amount of wastewater produced, it is then possible that all the water is reused and there is no effluent from a cleaning operation. Furthermore, the reuse allows for the capturing of the product residue that is left in the processing unit. This could account for substantial economic gain. Lastly, the amount of water that is used in product is reduced and water that would normally be discarded is now used in product, which also allows for some economic gain.

The formulations presented in this section are based on the type of wastewater reuse described above, where wastewater is produced from a cleaning operation and reused in product. However, due to the fact that product integrity is a high priority a number of conditions must be ensured. Firstly, the reuse of wastewater contaminated with a certain residue can only be reused in compatible product. In most instances, this is the same product as the residue. Secondly, wastewater containing different contaminants cannot be stored at the same time in the same storage vessel. Finally, the water used for a cleaning operation has to be of the same quality as water used in product.

Apart from the above conditions a number of assumptions have been made to simplify the operation. These assumptions are as follows.

- Once product is removed from a processing unit the unit is cleaned directly afterwards
- The ratio between water and other constituents in product is fixed
- The batch size and amount of water used for a cleaning operation is fixed for the zero effluent scheduling model.

The type of operation considered in the zero effluent methodology means that the amount of time points used for an operation has to increase. This is due to the fact that there is a processing step and a cleaning step associated with each batch of product. Normally two time points are used to describe a task in a unit. The first time point is used when the task commences in a unit and the second when the task terminates in a unit. In the type of operation considered in the zero effluent models, three time points are used. At the first time point the raw material processing task commences. The raw material processing step ends at the second time point, where the final product is removed and the cleaning operation commences. At the third time point the cleaning operation comes to an end and wastewater is produced.

The methodology deals with two types of problems, namely, the wastewater minimisation problem within a given plant structure and the plant synthesis problem. Each of these is dealt with in the form of two mathematical formulations. The first mathematical formulation deals with the scheduling of an existing operation as to produce near zero effluent. The second mathematical formulation deals with the synthesis of a batch plant which is capable of producing near zero effluent. The synthesis formulation and the scheduling formulation are intricately linked through a number of scheduling constraints and mass balance constraints that are common in both models. Each is described in subsequent subsections below. Both formulations are based on a common superstructure.

8.2 Superstructure Used in the Methodology

The superstructure used for both the zero effluent scheduling formulation and the synthesis formulation is given in Fig. 8.1. Figure 8.1a shows a unit operating in zero effluent mode. The raw materials used in such a unit is comprised of freshwater, directly recycled/reused water, water from storage and any other raw materials required. Furthermore, it is apparent from the figure that the water leaving after a cleaning operation is either directly reused in compatible product, sent to storage for later reuse or discarded as effluent. Figure 8.1b shows the overall plant superstructure of which each of the units in Fig. 8.1a form a part of. As one would notice from the plant superstructure, there are multiple storage vessels available for wastewater.

The constraints comprising the two formulations are described in two sections below. The first section deals with the constraints necessary for the zero effluent

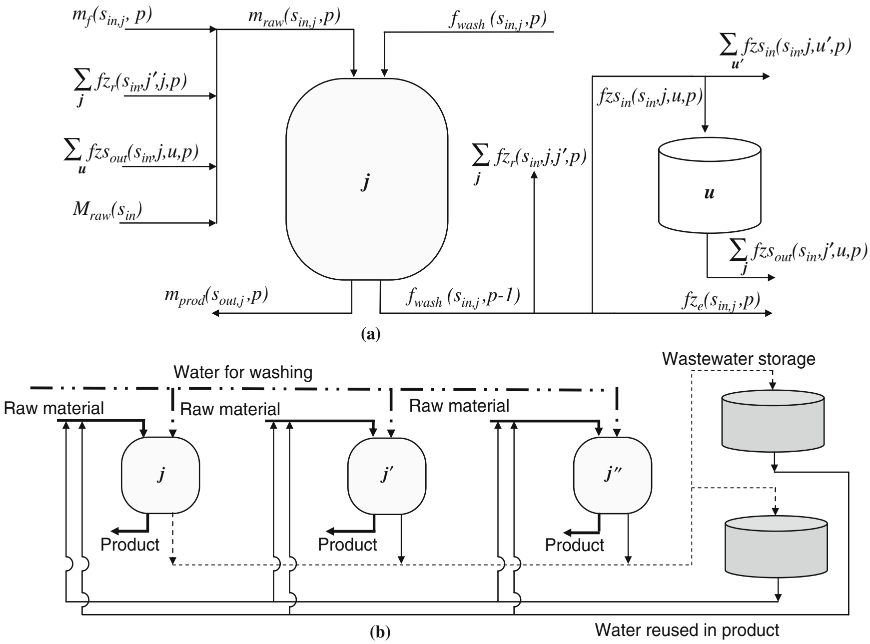


Fig. 8.1 Superstructure for zero effluent formulations (Gouws and Majozi, 2008)

scheduling formulation and the second section deals with the constraints that comprise the zero effluent synthesis formulation. A full description of the sets, variables and parameters used in both formulations can be found in the nomenclature list.

8.3 Zero Effluent Scheduling Model

The following sets, variables and parameters constitute the mathematical formulation for the nature of the problem presented in this chapter.

Sets

$$
\begin{aligned}
P &= \{p|p = \text{time point}\}\\
J &= \{j|j = \text{unit}\}\\
C &= \{c|c = \text{contaminant}\}\\
U &= \{u|u = \text{reusable water storage vessel}\}\\
S_{\text{in}} &= \{S_{\text{in}}|S_{\text{in}} = \text{input state into any unit}\}\\
S_{\text{out}} &= \{S_{\text{out}}|S_{\text{out}} = \text{output state from any unit}\}\\
S_{\text{in},j} &= \left\{S_{\text{in},j}|S_{\text{in},j} = \text{input state into unit } j\right\} \subseteq S_{\text{in}}\\
S_{\text{out},j} &= \left\{S_{\text{out},j}|S_{\text{out},j} = \text{output state from unit } j\right\} \subseteq S_{\text{out}}\\
S &= \{s|s\ \text{isastate}\} = s_{\text{in},j} \cup S_{\text{out},j}
\end{aligned}
$$

Continuous Variables

$czs\,(s_{\text{in}}, u, p)$	concentration of state s_{in} in storage vessel u at time point p
$f_{\text{water}}\left(s_{\text{in},j}, p\right)$	mass of water used for the cleaning operation in unit j at time point p
$fz_e\,(s_{\text{in}}, j, p)$	mass of water discarded as effluent time point p
$fz_r\left(s_{\text{in}}, j, j', p\right)$	mass of water recycled as raw material to unit j' from j at time point p
$fzs_{\text{in}}\,(s_{\text{in}}, j, u, p)$	mass of water to storage vessel u from unit j at time point p
$fzs_{\text{out}}\,(s_{\text{in}}, u, j, p)$	mass of water from storage vessel u to unit j at time point p
$m_{\text{bulk}}\left(s_{\text{in},j}, p\right)$	mass of raw material, other than water, from bulk storage to unit j at time point p
$m_f\left(s_{\text{in},j}, p\right)$	mass of freshwater used in product at time point p from unit j
$m_{\text{prod}}\left(s_{\text{out},j}, p\right)$	mass of product produced at time point p from unit j
$m_{\text{raw}}\left(s_{\text{in},j}, p\right)$	mass of raw materials into unit j at time point p
$qz\,(s_{\text{in}}, u, p)$	amount of water stored in storage vessel u at time point p
$t_p\left(s_{\text{out},j}, p\right)$	time at which unit j produces product s_{out} at time point p

$t_u\left(s_{\text{in},j},p\right)$	time at which unit j starts processing raw material s_{in} at time point p
$t_{pw}\left(s_{\text{in},j},p\right)$	time at which water is produced from the cleaning operation in unit j at time point p, containing product s_{in}
$t_{uw}\left(s_{\text{in},j},p\right)$	time at which water is used for the cleaning operation in unit j, at time point p, containing product s_{in}
$tz_r\left(s_{\text{in}},j,j',p\right)$	time at which water is recycled from unit j to unit j' at time point p
$tzs_{\text{in}}\left(s_{\text{in}},j,u,p\right)$	time at which water goes to storage vessel u from unit j at time point p
$tzs_{\text{out}}\left(s_{\text{in}},u,j,p\right)$	time at which water leaves storage vessel u to unit j at time point p
$\text{v}_{\text{unit}}(j)$	capacity of unit j
$\text{v}_{\text{stor}}(u)$	capacity of storage vessel u
$\Gamma_{\text{in}}(u,p)$	variable accounting for the summation of the binary variables entering a storage vessel
$\Gamma_{\text{out}}(u,p)$	variable accounting for the summation of the binary variables leaving a storage vessel

Binary Variables

$e_{\text{unit}}(j)$	existence binary variable showing the existence of unit j
$e_{\text{stor}}(u)$	existence binary variable showing the existence of storage vessel u
$y\left(s_{\text{in},j},p\right)$	binary variable showing usage of raw material s_{in} in unit j at time point p
$yz_r(s_{\text{in}},j,j',p)$	binary variable showing usage of recycle from unit j to unit j' at time point p
$yzs_{\text{in}}(s_{\text{in}},j,u,p)$	binary variable for water moving from unit j, at time point p, to storage vessel u
$yzs_{\text{out}}(s_{\text{in}},u,j,p)$	binary variable for water moving from storage vessel u to unit j at time point p

Parameters

C_{e}	effluent treatment cost
$C_{\text{out}}(s_{\text{in}},j)$	outlet concentration of state s_{in} in water from unit j
H	time horizon of interest
$F_{\text{wash}}(j)$	fixed mass of water used for a cleaning operation in unit j

$M_{\text{batch}}\left(s_{\text{in}}\right)$	size of batch using state s_{in}
$M_{\text{raw}}\left(s_{\text{in}}\right)$	mass of raw material, other than water, used in product
$M_{\text{res}}\left(s_{\text{in},j}\right)$	residue mass of state s_{in} in unit j
$M_{\text{water}}\left(s_{\text{in}}\right)$	fixed mass water used in product
$Q_z^o\left(s_{\text{in}},u\right)$	initial amount of water stored in storage vessel u
$Q_z^{\max}(u)$	maximum storage capacity of storage vessel u
$TS^{\max}$	maximum storage time of water in a storage vessel
$V_{\text{unit}}^{\min}$	minimum capacity of a processing unit
$V_{\text{unit}}^{\max}$	maximum capacity of a processing unit
$V_{\text{stor}}^{\min}$	minimum capacity of a storage vessel
$V_{\text{stor}}^{\max}$	maximum capacity of a storage vessel
α_{unit}	cost coefficient of a unit
α_{stor}	cost coefficient of a storage vessel
β_{unit}	cost coefficient of unit for the size of the unit
β_{stor}	cost coefficient of a storage vessel for the size of the vessel
ψ_{wash}	proportionality factor relating the amount of water used for a cleaning operation to the size of the processing unit
$\tau_p\left(s_{\text{in},j}\right)$	mean processing time of state s_{in} in unit j
$\tau_{\text{wash}}(j)$	mean processing time of the cleaning operation in unit j

It is first necessary to define the problem that is addressed in the formulation, as given in the following section.

8.3.1 *Problem Statement for the Zero Effluent Scheduling Formulation*

The problem addressed in the zero effluent scheduling formulation can be formally stated as follows.
Given,

(i) the recipe of each product, including the ratio of water to other components,
(ii) the average processing times of each product,
(iii) the quantity of water used in the cleaning operation,
(iv) the number of storage vessels, their capacities and the type of water that can be stored in each vessel,
(v) the compatibility of each product with the other products, and
(vi) the time horizon of interest,

determine the production schedule that will result in the generation of near zero effluent by exploiting the reuse of wastewater as part of product constituents.

The constraints that comprise the mathematical formulation are presented in two modules. The first deals with the mass balance constraints and the second the scheduling constraints.

8.3.2 Mass Balance Constraints

The first constraints that are dealt with in the zero effluent scheduling formulation are the unit mass balance constraints. These constraints account for the movement of mass between the various units and storage vessels.

Unit Mass Balances with Negligible Contaminant Mass in the Wastewater

Constraint (8.1) is a raw material mass balance into a unit. The amount of raw material into a unit is the sum of the directly reused water, freshwater, water from storage and any other raw materials required for the specific final product. Constraint (8.1) is the form of the raw material balance where the contaminant mass in the reused water is negligible. Constraint (8.1) for the case where the contaminant mass is not negligible will be given at a later stage. It is important to note that only compatible water can be reused in product. The reuse streams each contain information on the contaminant present in the water through the state indices in the variables describing the reuse flows.

$$\begin{aligned} m_{\mathrm{raw}}\left(s_{\mathrm{in},j},p\right) &= \sum_{j'} fz_r\left(s_{\mathrm{in}},j',j,p\right) + m_f\left(s_{\mathrm{in},j},p\right) + \sum_{u} fzs_{\mathrm{out}}\left(s_{\mathrm{in}},u,j,p\right) \\ &\quad + M_{\mathrm{raw}}\left(s_{\mathrm{in}}\right) y\left(s_{\mathrm{in},j},p\right) \\ &\forall\ s_{\mathrm{in}} \in S_{\mathrm{in}},\ j,j' \in J, p \in P,\ u \in U, s_{\mathrm{in},j} \in S_{\mathrm{in},j} \end{aligned} \tag{8.1}$$

The model is based on fixed batch size, as mentioned previously. The batch size is fixed to the appropriate level in constraint (8.2). A product mass balance over a unit is given in constraint (8.3). This constraint states that the mass product produced at a time point is the amount of raw material used at the previous time point less the residue left in the unit.

$$m_{\mathrm{raw}}\left(s_{\mathrm{in},j},p\right) = M_{\mathrm{batch}}\left(s_{\mathrm{in}}\right) y\left(s_{\mathrm{in},j},p\right),\ \forall\ s_{\mathrm{in},j} \in S_{\mathrm{in},j}, p \in P, s_{\mathrm{in}} \in S_{\mathrm{in}} \tag{8.2}$$

$$\begin{aligned} m_{\mathrm{prod}}\left(s_{\mathrm{out},j},p\right) &= m_{\mathrm{raw}}\left(s_{\mathrm{in},j},p-1\right) - M_{\mathrm{res}}\left(s_{\mathrm{in},j}\right) y\left(s_{\mathrm{in},j},p-1\right), \\ &\forall\ j \in J, s_{\mathrm{in},j} \in S_{\mathrm{in},j}, s_{\mathrm{out},j} \in S_{\mathrm{out},j}, p \in P, p > p_1 \end{aligned} \tag{8.3}$$

The amount of water used for a cleaning operation is assumed to be fixed. The amount of water used for a cleaning operation is thus defined in constraint (8.4). The water leaving a cleaning operation can either be directly reused in a subsequent batch of compatible product, sent to storage for later reuse or discarded. This is captured in constraint (8.5).

$$f_{\text{wash}}\left(s_{\text{in},j},p\right)=F_{\text{wash}}\left(j\right)y\left(s_{\text{in},j},p-1\right),\ \forall\ s_{\text{in},j}\in S_{\text{in},j},p\in P,p>p_1,j\in J \quad (8.4)$$

$$f_{\text{wash}}\left(s_{\text{in},j},p-1\right)=\sum_{j'}fz_r\left(s_{\text{in}},j,j',p\right)+fz_e\left(s_{\text{in},j},p\right)+\sum_{u}fzs_{\text{in}}\left(s_{\text{in}},j,u,p\right),$$
$$\forall\ j,j'\in J,s_{\text{in}}\in S_{\text{in}},p\in P,p>p_1,\ u\in U,s_{\text{in},j}\in S_{\text{in},j} \quad (8.5)$$

Constraints (8.6) and (8.7) ensure that the amount of water reused in product directly and indirectly is less than the amount of water used in product, respectively. The mass of water used in product is fixed due to the ratio of water and other raw materials being fixed and the batch size being fixed.

$$fzs_{\text{out}}\left(s_{\text{in}},j,u,p\right)\le M_{\text{water}}\left(s_{\text{in}}\right)yzs_{\text{out}}(s_{\text{in}},j,u,p),$$
$$\forall\ j\in J,u\in U,s_{\text{in}}\in S_{\text{in}},p\in P,j\in J \quad (8.6)$$

$$fz_r\left(s_{\text{in}},j',j,p\right)\le M_{\text{water}}\left(s_{\text{in}}\right)yz_r(s_{\text{in}},j',j,p),\ \forall\ s_{\text{in}}\in S_{\text{in}},p\in P,j,j'\in J \quad (8.7)$$

The amount of water that is sent to a storage vessel must be less than the capacity of the storage vessel. This is ensured through constraint (8.8).

$$fzs_{\text{in}}\left(s_{\text{in}},j,u,p\right)\le Q_z^{\max}\left(u\right)yzs_{\text{in}}(s_{\text{in}},j,u,p),$$
$$\forall\ s_{\text{in}}\in S_{\text{in}},p\in P,j\in J,u\in U \quad (8.8)$$

Due to the fact that the contaminant mass in the wastewater is assumed to be negligible at this stage, one does not have to consider contaminant balances.

Storage Mass Balances with Negligible Contaminant Mass in the Wastewater

Storage mass balances also have to be included to capture the movement of mass to and from the various storage vessels. The first mass balance considered is a water balance over a storage vessel. The amount of water stored at a time point is the amount of water stored from the previous time point and difference between the amount of water entering and leaving a storage vessel at that time point. This is given in constraint (8.9). The water balance for the first time point is given in constraint (8.10). The amount of water stored in a storage vessel at any time point must be less than the capacity of the storage vessel. This is ensured through constraint (8.11).

$$qz\left(s_{\text{in}},u,p\right)=qz\left(s_{\text{in}},u,p-1\right)+\sum_{j}fzs_{\text{in}}\left(s_{\text{in}},j,u,p\right)-\sum_{j}fzs_{\text{out}}\left(s_{\text{in}},j,u,p\right),$$
$$\forall\ j\in J,s_{\text{in}}\in S_{\text{in}},p\in P,p>p_1,u\in U \quad (8.9)$$

$$qz\left(s_{\text{in}},u,p_1\right)=Q_z^0\left(s_{\text{in}},u\right)-\sum_{j}fzs_{\text{out}}\left(s_{\text{in}},u,j,p_1\right),\ \ \forall\ j\in J,s_{\text{in}}\in S_{\text{in}},u\in U \quad (8.10)$$

$$qz\left(s_{\text{in}},u,p\right)\le Q_z^{\max}\left(u\right),\ \forall\ s_{\text{in}}\in S_{\text{in}},p\in P,u\in U \quad (8.11)$$

As with the mass balances over a unit, contaminant balances around a storage vessel are not required due to the contaminant mass being negligible at this stage.

The constraints given above would suffice if the operation under consideration was a continuous process. However, since the operation under consideration is a batch process, constraints have to be presented to account for the discontinuous nature of the process.

8.3.3 Scheduling Constraints

The first scheduling constraints considered are binary variable constraints governing wastewater reuse. These constraints ensure the correct reuse of water, since water is only reused in distinct amounts at certain points in the time horizon.

Binary Variable Constraints for Wastewater Reuse

Binary variable constraints are defined to ensure water is reused, both directly and indirectly, under the correct conditions. Water can only be directly reused if the sink unit, i.e. the unit receiving the water, is operating at that time point. However, it is not a prerequisite for a unit to receive reuse water to operate. This is all captured in constraint (8.12). Furthermore, water can only be directly reused if the source process, i.e. the cleaning process, has finished at that time point, given in constraint (8.13). The binary variable showing the usage of a unit is used in this constraint due to the fact that the cleaning operation always occurs directly after the raw material processing operation. Constraint (8.13) also states that a cleaning operation can occur without direct reuse taking place.

$$yz_r\left(s_{\text{in}},j,j',p\right) \le y\left(s_{\text{in},j'},p\right),\ \ \forall\, j',j \in J, s_{\text{in},j'} \in S_{\text{in},j}, s_{\text{in}} \in S_{\text{in}}, p \in P, \qquad (8.12)$$

$$yz_r\left(s_{\text{in}},j,j',p\right) \le y\left(s_{\text{in},j},p-2\right),\ \ \forall\, j',j \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j} \in S_{\text{in},j}, p \in P, \qquad (8.13)$$

Water sent to a storage vessel can only do so once two criteria are fulfilled. Firstly, water can only be stored if a cleaning operation has been completed at that time point. Secondly, water can only be sent to a storage vessel provided the water already in the storage vessel contains the same or compatible contaminants. Irrespective of whether the contaminant mass is negligible or not, water stored in a storage vessel must all contain the same contaminants or compatible contaminants. This ensures product integrity is not compromised. Constraint (8.14) ensures that if water is sent to a storage vessel the cleaning operation in the source unit has occurred and come to an end. The type of water entering a storage vessel is controlled by dedicating certain storage vessels to the storage of wastewater containing certain types of contaminants. Constraint (8.14) also states that a cleaning operation can occur without sending water to storage.

$$yzs_{\text{in}}\left(s_{\text{in}},j,u,p\right) \le y\left(s_{\text{in},j},p-2\right),\ \ \forall\, j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P, p > p_2, u \in U \qquad (8.14)$$

Water sent to a unit from a storage vessel for reuse can only occur if the unit is operating at that time point and the water and product are compatible. This is ensured through constraint (8.15). A unit can also operate without receiving water from storage, which is also ensured in constraint (8.15).

$$yzs_{\mathrm{out}}\left(s_{\mathrm{in}},u,j,p\right)\leq y\left(s_{\mathrm{in},j},p\right),\ \ \forall j\in J,s_{\mathrm{in},j}\in S_{\mathrm{in},j},p\in P,u\in U \tag{8.15}$$

Due to the nature of the process constraints have to be derived that capture the essence of time. The first of these considered deal with the scheduling of the tasks in a unit.

Unit Scheduling Constraints

The first constraint considered is a task duration constraint, given in constraint (8.16). This constraint states that the time at which a task ends and product is produced is the starting time of task and the duration of the task.

$$\begin{gathered} t_p\left(s_{\mathrm{out},j},p\right)=t_u\left(s_{\mathrm{in},j},p-1\right)+\tau_p\left(s_{\mathrm{in},j}\right)y\left(s_{\mathrm{in},j},p-1\right)\\ \forall j\in J,s_{\mathrm{in},j}\in S_{\mathrm{in},j},s_{\mathrm{out},j}\in S_{\mathrm{out},j},p\in P,p>p_1 \end{gathered} \tag{8.16}$$

Directly after product has been produced from a task in a unit, the cleaning operation in the unit begins. Constraints (8.17) and (8.18) ensure that the time at which product is produced and the time at which a cleaning operation begins in a unit correspond to the same time.

$$\begin{gathered} t_{\mathrm{uw}}\left(s_{\mathrm{in},j},p\right)\geq t_p\left(s_{\mathrm{out},j},p\right)-H\left(1-y\left(s_{\mathrm{in},j},p-1\right)\right)\\ \forall j\in J,s_{\mathrm{in},j}\in S_{\mathrm{in},j},s_{\mathrm{out},j}\in S_{\mathrm{out},j},p\in P,p\geq p_1 \end{gathered} \tag{8.17}$$

$$\begin{gathered} t_{\mathrm{uw}}\left(s_{\mathrm{in},j},p\right)\leq t_p\left(s_{\mathrm{out},j},p\right)-H\left(1-y\left(s_{\mathrm{in},j},p-1\right)\right)\\ \forall j\in J,s_{\mathrm{in},j}\in S_{\mathrm{in},j},s_{\mathrm{out},j}\in S_{\mathrm{out},j},p\in P,p\geq p_1 \end{gathered} \tag{8.18}$$

The starting and ending times of a cleaning operation in a unit are related through the cleaning operation duration constraint given in constraint (8.19). This constraint is similar to the task duration constraint given earlier.

$$\begin{gathered} t_{\mathrm{pw}}\left(s_{\mathrm{in},j},p\right)=t_{uw}\left(s_{\mathrm{in},j},p-1\right)+\tau_{\mathrm{wash}}\left(j\right)y\left(s_{\mathrm{in},j},p-2\right)\\ \forall j\in J,s_{\mathrm{in},j}\in S_{\mathrm{in},j},p\in P,p>p_2 \end{gathered} \tag{8.19}$$

Due to the fact that it is assumed that a washout occurs directly after product is removed, two consecutive batches of product will be separated by a washout. In terms of time points, a unit can only start processing a batch two time points after the first batch starts. This is due to the fact that three time points are used to describe the batch processing and unit cleaning operations. Furthermore, the assumption that a cleaning operation follows product removal negates the need for a separate binary variable to represent the cleaning operation in a unit. Therefore, constraint (8.20) is included to ensure that two product producing tasks do not occur in consecutive time points.

$$y\left(s_{\mathrm{in},j},p\right)+y\left(s_{\mathrm{in},j'},p-1\right)\leq 1\quad\forall j\in J,s_{\mathrm{in},j},s_{\mathrm{in},j'}\in S_{\mathrm{in},j},p\in P,p\geq p_1, \tag{8.20}$$

Constraint (8.21) ensures that a unit can only start processing a batch once the unit has finished the cleaning operation. Constraints (8.22) and (8.23) ensure that if a unit starts operating or finishes operating at a later time point, the time at which this occurs corresponds to a later time in the time horizon.

$$t_u\left(s_{\text{in},j},p\right) \geq t_{pw}\left(s_{\text{in},j'},p'\right) - H\left(2 - y\left(s_{\text{in},j},p\right) - y\left(s_{\text{in},j'},j,p'\right)\right), \quad \forall\, j \in J, s_{\text{in},j'}, s_{\text{in},j} \in S_{\text{in},j}, p \in P, p \geq p', \tag{8.21}$$

$$t_u\left(s_{\text{in},j},p\right) \geq t_u\left(s_{\text{in},j'},p'\right) - H\left(2 - y\left(s_{\text{in},j},p\right) - y\left(s_{\text{in},j'},p'\right)\right), \quad \forall\, j \in J, s_{\text{in},j'}, s_{\text{in},j} \in S_{\text{in},j}, p, p' \in P, p \geq p', \tag{8.22}$$

$$t_p\left(s_{\text{in},j},p\right) \geq t_p\left(s_{\text{in},j'},p'\right) - H\left(2 - y\left(s_{\text{in},j},p\right) - y\left(s_{\text{in},j'},j,p'\right)\right), \quad \forall\, j \in J, s_{\text{in},j}, s_{\text{in},j'} \in S_{\text{in},j}, p, p' \in P, p \geq p' \tag{8.23}$$

Scheduling constraints also have to be derived to ensure the time at which wastewater reuse occurs is correct within the time horizon.

Direct Recycle/Reuse Scheduling

Due to the discontinuous availability of wastewater for reuse, constraints have to be formulated that ensure the correct timing of water reuse when it occurs. Constraints (8.24) and (8.25) ensure that the time at which wastewater is produced and the time at which the wastewater is recycle/reused correspond to the same time.

$$tz_r\left(s_{\text{in}},j,j',p\right) \leq t_{\text{pw}}\left(s_{\text{in},j},p\right) + H\left(1 - yz_r\left(s_{\text{in}},j,j',p\right)\right), \quad \forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{8.24}$$

$$tz_r\left(s_{\text{in}}j,j',p\right) \geq t_{\text{pw}}\left(s_{\text{in},j},p\right) - H\left(1 - yz_r\left(s_{\text{in}},j,j',p\right)\right), \quad \forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{8.25}$$

Constraints (8.26) and (8.27) ensure that the time at which wastewater is recycled/reused to a unit and the starting time of the task in the unit receiving the wastewater correspond to the same time.

$$tz_r\left(s_{\text{in}},j,j',p\right) \leq t_u\left(s_{\text{in},j'},p\right) + H\left(1 - yz_r\left(s_{\text{in}},j,j',p\right)\right), \quad \forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j'} \in S_{\text{in},j}, p \in P \tag{8.26}$$

$$tz_r\left(s_{\text{in}},j,j',p\right) \geq t_u\left(s_{\text{in}},j',p\right) - H\left(1 - yz_r\left(s_{\text{in}},j,j',p\right)\right), \quad \forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j'} \in S_{\text{in},j}, p \in P \tag{8.27}$$

As with the direct reuse of wastewater, scheduling of indirect reuse of water is also necessary.

Indirect Reuse Scheduling

Constraints also have to be derived to ensure the correct timing of streams to and from the various storage vessels. The time at which wastewater is sent to a storage vessel and the time at which wastewater is produced from a cleaning operation must correspond to the same time. This is ensured through constraints (8.28) and (8.29).

$$tzs_{\text{in}}\left(s_{\text{in}},j,u,p\right) \geq t_{\text{pw}}\left(s_{\text{in},j},p\right) - H\left(1 - yzs_{\text{in}}\left(s_{\text{in},j},u,p\right)\right),\quad \forall\, j \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j} \in S_{\text{in},j}, p \in P, u \in U \tag{8.28}$$

$$tzs_{\text{in}}\left(s_{\text{in}},j,u,p\right) \leq t_{\text{pw}}\left(s_{\text{in},j},p\right) + H\left(1 - yzs_{\text{in}}\left(s_{\text{in}},j,u,p\right)\right),\quad \forall\, j \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j} \in S_{\text{in},j}, p \in P, u \in U \tag{8.29}$$

Constraints (8.30) and (8.31) ensure that the time at which water is sent to a unit and the starting time of the task in the unit receiving the water corresponds to the same time.

$$tzs_{\text{out}}\left(s_{\text{in}},u,j,p\right) \geq t_u\left(s_{\text{in},j},p\right) - H\left(1 - yzs_{\text{out}}\left(s_{\text{in}},u,j,p\right)\right),\quad \forall\, j \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j} \in S_{\text{in},j}, p \in P, u \in U \tag{8.30}$$

$$tzs_{\text{out}}\left(s_{\text{in}},u,j,p\right) \leq t_u\left(s_{\text{in},j},p\right) - H\left(1 - yzs_{\text{out}}\left(s_{\text{in}},u,j,p\right)\right),\quad \forall\, j \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j} \in S_{\text{in},j}, p \in P, u \in U \tag{8.31}$$

Apart from the scheduling of the streams entering and exiting a storage vessel with regard to the source and sink units, the scheduling of multiple streams entering and leaving a storage vessel has to be considered. If water is leaving a storage vessel at a later time point, this must occur at a later actual time in the time horizon. This is ensured through constraint (8.32). Constraints (8.33) and (8.34) ensure that if two streams are entering a storage vessel at a time point, the time at which each enters corresponds to the same absolute time in the time horizon.

$$tzs_{\text{out}}\left(s_{\text{in}},u,j,p\right) \geq tzs_{\text{out}}\left(s_{\text{in}},u,j',p'\right) - H\left(2 - yzs_{\text{out}}\left(s_{\text{in}},u,j,p\right) - yzs_{\text{out}}\left(s_{\text{in}},u,j',p'\right)\right),\quad \forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, p,p' \in P, p' < p, u \in U, \tag{8.32}$$

$$tzs_{\text{out}}\left(s_{\text{in}},u,j,p\right) \geq tzs_{\text{out}}\left(s_{\text{in}},u,j',p\right) - H\left(2 - yzs_{\text{out}}\left(s_{\text{in}},u,j,p\right) - yzs_{\text{out}}\left(s_{\text{in}},u,j',p\right)\right),\quad \forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, p \in P, u \in U \tag{8.33}$$

$$tzs_{\text{out}}\left(s_{\text{in}},u,j,p\right) \leq tzs_{\text{out}}\left(s_{\text{in}},u,j',p\right) + H\left(2 - yzs_{\text{out}}\left(s_{\text{in}},u,j,p\right) - yzs_{\text{out}}\left(s_{\text{in}},u,j',p\right)\right),\quad \forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, p \in P, u \in U \tag{8.34}$$

Similar constraints hold for multiple water streams entering a storage vessel. Constraint (8.35) ensures that water entering storage at a later time point does so at a later actual time in the time horizon. Constraints (8.36) and (8.37) ensure that the times at which two streams leave a storage vessel, at the same time point, correspond with the same time.

$$tzs_{\text{in}}\left(s_{\text{in}},j,u,p\right) \geq tzs_{\text{in}}\left(s_{\text{in}},j',u,p'\right) - H\left(2 - yzs_{\text{in}}\left(s_{\text{in}},j,u,p\right) - yzs_{\text{in}}\left(s_{\text{in}},j',u,p'\right)\right),\quad \forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, p,p' \in P, p > p', u \in U, \tag{8.35}$$

$$tzs_{\text{in}}\left(s_{\text{in}},j,u,p\right) \geq tzs_{\text{in}}\left(s_{\text{in}},j',u,p\right) - H\left(2 - yzs_{\text{in}}\left(s_{\text{in}},j,u,p\right) - yzs_{\text{in}}\left(s_{\text{in}},j',u,p\right)\right),\quad \forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, p \in P, u \in U \tag{8.36}$$

$$tzs_{\text{in}}(s_{\text{in}},j,u,p) \le tzs_{\text{in}}\left(s_{\text{in}},j',u,p\right) + H\left(2 - yzs_{\text{in}}(s_{\text{in}},j,u,p) - yzs_{\text{in}}\left(s_{\text{in}},j',u,p\right)\right),$$
$$\forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, p \in P, u \in U \quad (8.37)$$

Constraints (8.38) - (8.40) are constraints that deal with the scheduling of streams to and from a storage vessel. If water leaves a storage vessel at a time point after the time point at which the water entered the vessel, then the time at which this happens must occur at a later absolute time in the time horizon. This is given in constraint (8.38). The time at which a stream leaves a storage vessel and the time at which water enters a storage vessel must coincide, provided the two streams enter at the same time point. This is ensured through constraints (8.39) and (8.40).

$$tzs_{\text{out}}(s_{\text{in}},u,j,p) \ge tzs_{\text{in}}\left(s_{\text{in}},j',u,p'\right) - H\left(2 - yzs_{\text{out}}(s_{\text{in}},u,j,p) - yzs_{\text{in}}\left(s_{\text{in}},j',u,p'\right)\right),$$
$$\forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, p,p' \in P, p > p', u \in U, \quad (8.38)$$

$$tzs_{\text{out}}(s_{\text{in}},u,j,p) \ge tzs_{\text{in}}\left(s_{\text{in}},j',u,p\right) - H\left(2 - yzs_{\text{out}}(s_{\text{in}},u,j,p) - yzs_{\text{in}}\left(s_{\text{in}},j',u,p\right)\right),$$
$$\forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, p \in P, u \in U \quad (8.39)$$

$$tzs_{\text{out}}(s_{\text{in}},u,j,p) \le tzs_{\text{in}}\left(s_{\text{in}},j',u,p\right) + H\left(2 - yzs_{\text{out}}(s_{\text{in}},u,j,p) - yzs_{\text{in}}\left(s_{\text{in}},j',u,p\right)\right),$$
$$\forall\, j,j' \in J, s_{\text{in}} \in S_{\text{in}}, p \in P, u \in U \quad (8.40)$$

The final storage vessel scheduling constraint considered deals with the maximum amount of time wastewater can spend in storage. This type of consideration is necessary if there are possible contamination risks if the water is stored too long, e.g. microbial growth in the water. Constraint (8.41) ensures that the difference between the time at which water is sent to storage and the time at which water leaves storage is less than a predefined maximum. Constraints (8.42), (8.43) and (8.44) are added to ensure that water leaves a storage vessel at the time point following that at which the water entered.

$$\begin{aligned} tzs_{\text{out}}(s_{\text{in}},u,j,p) - tzs_{\text{in}}(s_{\text{in}},j,u,p-1) &\le TS^{\max} \\ &+ H\left(2 - yzs_{\text{out}}(s_{\text{in}},u,j,p) - yzs_{\text{in}}(s_{\text{in}},j,u,p-1)\right), \\ &\forall\, j \in J, s_{\text{in}} \in S_{\text{in}}, p \in P, p > p_1, u \in U \end{aligned} \quad (8.41)$$

$$\Gamma_{\text{in}}(u,p) = \sum_{j=1}^{J} \sum_{p=p_1}^{p' \le p-1} yzs_{\text{in}}\left(s_{\text{in}},j,u,p'\right), \forall\, s_{\text{in}} \in S_{\text{in}}, j \in J, p,p' \in P, u \in U \quad (8.42)$$

$$\Gamma_{\text{out}}(u,p) = \sum_{j=1}^{J} \sum_{p=p_1}^{p' \le p} yzs_{\text{out}}\left(s_{\text{in}},u,j,p'\right), \forall\, s_{\text{in}} \in S_{\text{in}}, j \in J, p,p' \in P, u \in U \quad (8.43)$$

$$\Gamma_{\text{out}}(u,p) = \Gamma_{\text{out}}(u,p), \forall\, p \in P, p > p_1, u \in U \quad (8.44)$$

Time Horizon Constraints

The final scheduling constraints considered are the time horizon constraints. These constraints ensure that each event occurs within the time horizon of interest and are given in constraints (8.45), (8.46), (8.47), (8.48), (8.49), (8.50) and (8.51).

$$tzs_{\text{in}}\left(s_{\text{in}}, j, u, p\right) \leq H, \quad \forall j \in J, s_{\text{in}} \in S_{\text{in}}, p \in P, u \in U \tag{8.45}$$

$$tzs_{\text{out}}\left(s_{\text{in}}, u, j, p\right) \leq H, \quad \forall j \in J, s_{\text{in}} \in S_{\text{in}}, p \in P, u \in U \tag{8.46}$$

$$t_u\left(s_{\text{in},j}, p\right) \leq H, \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{8.47}$$

$$t_p\left(s_{\text{out},j}, p\right) \leq H, \quad \forall j \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \tag{8.48}$$

$$tz_r\left(s_{\text{in}}, j, j', p\right) \leq H, \quad \forall j, j' \in J, p \in P, s_{\text{in}} \in S_{\text{in}} \tag{8.49}$$

$$t_{\text{uw}}\left(s_{\text{in},j}, p\right) \leq H, \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{8.50}$$

$$t_{\text{pw}}\left(s_{\text{in},j}, p\right) \leq H, \quad \forall j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{8.51}$$

8.3.4 Additional Constraints for Significant Contaminant Mass in the Wastewater

If the assumption that the contaminant mass in the wastewater is relaxed, then the additional raw material in the form of the contaminant mass has to be accounted for. The wastewater in this case not only supplements the water in the raw material, but also any other raw materials used in product formulation. The raw material balance given in constraint (8.1) is reformulated to account for the additional raw material source. Constraint (8.1) is split into a water balance and a raw material balance for the other components required in product formulation. The water balance is given in constraint (8.52). The balance for the other components used in the product formulation is given in constraint (8.53). Due to the fixed ratio of water and other components in product formulation and the fixed batch size, the amount of water and the amount of other components are fixed. Therefore, in constraints (8.52) and (8.53) the amount of water and amount of other raw material is fixed. The water balance, in constraint (8.52), states that the amount of water used in product is comprised of freshwater, water from storage and directly recycle/reused water. Constraint (8.53), the mass balance for the other components, states that the mass of other components used for product is the mass from bulk storage, the mass in directly recycled/reused water and the mass in water from storage.

$$M_{\text{water}}(s_{\text{in}})\, y\left(s_{\text{in},j}, p\right) = m_f\left(s_{\text{in},j}, p\right) + \sum_{j'} fz_r\left(s_{\text{in}}, j', j, p\right) + \sum_{u} fzs_{\text{out}}(s_{\text{in}}, u, j, p),$$
$$\forall\, j, j' \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j} \in S_{\text{in},j}, p \in P, u \in U \tag{8.52}$$

$$\begin{aligned} M_{\text{raw}}(s_{\text{in}})\, y\left(s_{\text{in},j}, p\right) = {} & m_{\text{bulk}}\left(s_{\text{in},j}, p\right) + \sum_{j'} C_{\text{out}}\left(s_{\text{in}}, j'\right) fz_r\left(s_{\text{in}}, j', j, p\right) \\ & + \sum_{u} czs\,(s_{\text{in}}, u, p) fzs_{\text{out}}(s_{\text{in}}, u, j, p), \end{aligned} \tag{8.53}$$
$$\forall\, j, j' \in J, s_{\text{in}} \in S_{\text{in}}, s_{\text{in},j} \in S_{\text{in},j}, p \in P, u \in U$$

It must be noted that the outlet concentration of wastewater generated by the cleaning operation is fixed. This is due to the amount of water used for a cleaning operation being fixed and the mass load also being fixed.

Due to the elevated contaminant mass load a contaminant balance also has to be included in the wastewater storage constraints. This is given in constraint (8.54), where the contaminant mass present in a storage vessel at a time point is the contaminant mass from the previous time point and the difference between the contaminant mass entering and exiting the vessel. Each storage vessel is assumed to be ideally mixed, hence the concentration within the vessel is uniform.

$$\begin{aligned} czs\,(s_{\text{in}}, u, p)\, q\,(s_{\text{in}}, u, p) = {} & czs\,(s_{\text{in}}, u, p-1)\, q\,(s_{\text{in}}, u, p-1) \\ & + \sum_{j} C_{\text{out}}(s_{\text{in}}, j) fzs_{\text{in}}(s_{\text{in}}, j, u, p) \\ & - \sum_{j} czs\,(s_{\text{in}}, u, p) fzs_{\text{out}}(s_{\text{in}}, u, j, p), \end{aligned} \tag{8.54}$$
$$\forall\, j \in J, s_{\text{in}} \in S_{\text{in}}, p \in P, p > p_1, u \in U$$

The final consideration in the formulation is the objective function.

8.3.5 Objective Function

The objective function can take two forms depending on the production information given. If the required production is known, then the objective is the minimisation of effluent. If this is not the case then the objective function takes the form of a profit function, where profit is dependent on the revenue from product, the cost of the raw material and the treatment costs of the effluent.

The constraints derived above complete the zero effluent scheduling formulation for both the situation where the contaminant mass is negligible and where it is not negligible.

A natural progression from the scheduling of zero effluent operations is the derivation of a formulation that synthesises batch plants operating in the zero effluent mode of operation. The problem to be solved is slightly different to the general scheduling problem addressed in the formulation presented previously.

8.4 Zero Effluent Plant Synthesis Formulation

The zero effluent synthesis formulation not only determines the optimal number and size of the storage and processing units, but also determines the schedule that will allow the resulting plant to operate in a near zero effluent operation. This is beneficial since scheduling considerations are taken into account during the plant synthesis phase. The exact problem considered in this formulation is given in the following section.

8.4.1 Problem Statement for Synthesis Formulation

The zero effluent synthesis formulation addresses the following problem.
Given,

(i) the required production over a time horizon,
(ii) the recipe of each product, including the ratio of water to other components,
(iii) the maximum number of processing units and storage vessels,
(iv) the minimum and maximum capacity of the possessing units and storage vessels,
(v) the relation between the amount of water for cleaning and size of processing unit, and
(vi) the compatibility of each product,

determine the number and size of processing units and storage vessels that will minimise the cost of the processing plant. The cost of the processing plant is dependent on the number and size of the processing units and storage vessels and the cost of the effluent generated by the plant.

8.4.2 Constraints Considered in the Formulation

The synthesis model is based on the zero effluent scheduling formulation with the addition of design specific constraints. The scheduling constraints used in the synthesis formulation are identical to those used in the scheduling formulation and the mass balance constraints are virtually identical except for a few minor changes in the limiting capacities, amount of water and amount of product. These changes will be discussed below.

The first constraints that are added in the synthesis formulation are existence constraints. In the synthesis formulation an existence binary variable is defined for each processing unit and each storage vessel. If a storage vessel exists the binary variable will assume the value of 1, if it does not exist the binary variable will assume the value of 0. This is the same for a processing unit. If a processing unit is used at any point within the time horizon, than unit must then exist. This is dealt with in constraint (8.55). This constraint states that if the binary variable showing the usage of a unit assumes the value of 1 at any time point then the existence binary variable will take the value of 1. A similar constraint holds for a storage vessel. If water is sent to a storage vessel, then the storage vessel must exist. This is given in constraint (8.56).

$$e_{\text{unit}}(j) \geq y\left(s_{\text{in},j}, p\right), \forall\, j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{8.55}$$

$$e_{\text{store}}(u) \geq yzs_{\text{in}}(s_{\text{in}}, j, u, p), \forall\, j \in J, s_{\text{in}} \in S_{\text{in}}, p \in P, u \in U \tag{8.56}$$

The capacity of each processing unit and storage vessel is an optimisation variable that must be bound to a minimum and maximum. The capacity of each processing unit and each storage vessel is limited to their respective maximum and minimum using constraints (8.57) and (8.58), respectively.

$$e_{\text{unit}}(j)\, V_{\text{unit}}^{\min} \leq \text{v}_{\text{unit}}(j) \leq e_{\text{unit}}(j)\, V_{\text{unit}}^{\max}, \forall\, j \in J \tag{8.57}$$

$$e_{\text{stor}}(u)\, V_{\text{stor}}^{\min} \leq \text{v}_{\text{stor}}(u) \leq e_{\text{stor}}(u)\, V_{\text{stor}}^{\max}, \forall\, u \in U \tag{8.58}$$

The first minor change to the mass balance constraints from the scheduling formulation is found in constraint (8.2), which defines the size of a batch. In the synthesis formulation, the batch size is determined by the optimal size of the processing unit. Due to this being a variable, constraint (8.2) is reformulated to reflect this and is given in constraint (8.59). The nonlinearity present in constraint (8.59) is linearised exactly using Glover transformation (1975) as presented in Chapter 4.

$$m_{\text{raw}}\left(s_{\text{in},j}, p\right) = \text{v}_{\text{unit}}(j)\, y\left(s_{\text{in},j}, p\right), \forall\, j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{8.59}$$

The amount of water used for a cleaning operation is also now variable and is dependent on the size of the processing unit. Constraint (8.6) is reformulated to account for this and takes the form of constraint (8.60). Constraint (8.60) shows that the amount of water used is related to the size of the processing unit through a proportionality factor. Once again there is a nonlinear term present in constraint (8.60), which is linearised exactly using a Glover transformation (1975).

$$f_{\text{wash}}\left(s_{\text{in},j}, p\right) = \psi_{\text{wash}} \text{v}_{\text{unit}}(j)\, y\left(s_{\text{in},j}, p\right), \forall\, j \in J, s_{\text{in},j} \in S_{\text{in},j}, p \in P \tag{8.60}$$

The capacity of a storage vessel is also variable. The amount of water stored in a vessel is bounded by capacity of the storage vessel. This is given in constraint (8.61), which is the reformulated form of constraint (8.11).

$$qz(s_{\text{in}}, u, p) \leq \text{v}_{\text{stor}}(u), \forall\, s_{\text{in}} \in S_{\text{in}}, u \in U, p \in P \tag{8.61}$$

The objective function used in the zero effluent synthesis formulation is the minimisation of cost. The cost function is defined in constraint (8.62). The cost in constraint (8.62) is the sum of the cost of the processing units, cost of the storage vessels and the cost of the treatment of the effluent produced.

$$\sum_{j\in J}(\alpha_{\text{unit}}e_{\text{unit}}(j)+\beta_{\text{unit}}v_{\text{unit}}(j))+\sum_{u\in U}(\alpha_{\text{stor}}e_{\text{stor}}(u)+\beta_{\text{stor}}v_{\text{stor}}(u))+\sum_{s_{\text{in},j}\in S_{\text{in},j},p\in P}C_e f_e\left(s_{\text{in},j},p\right), \tag{8.62}$$

The above models were applied to two illustrative examples and applied to an industrial case study. The illustrative examples are discussed below.

8.5 Illustrative Examples Using the Zero Effluent Mode of Operation

Two illustrative examples are presented. The first deals with the scheduling of a small batch process operating in zero effluent mode and the second deals with the synthesis of a small batch operation.

8.5.1 First Illustrative Example

The operation considered produces two products from two mixing vessels, with each mixing vessel dedicated to the mixing of a distinct product, i.e., mixer one produces product 1 and mixer two produces product 2. Each product has common raw material, apart from water, that is produced by a reactor. Product, therefore, comprises of a certain amount of raw material from the reactor, water and other raw material specific to the product. Once each product has been removed from the mixer, the mixer is cleaned. The wastewater is of such a nature that it can be reused as constituents in a subsequent batch of the same product as the residue in the wastewater, i.e., wastewater from mixer one, containing product 1 as a contaminant, can be reused in a subsequent batch of product 1. Wastewater containing product 1 as a contaminant and wastewater containing product 2 as a contaminant cannot be mixed. Two wastewater storage vessels are therefore used, with each dedicated to the storage of wastewater containing a specific contaminant. A process diagram is given in Fig. 8.1.

It is assumed in this example that the contaminant mass in the wastewater is negligible. This assumption, therefore, leads to the resulting model being a MILP.

The composition of each product is given in Table 8.1. The batch size of each product is fixed at 2000 kg. Important to note in Table 8.1 that the ratio between the various raw materials in each product is fixed.

The average processing time for the reactor to produce a batch of product is 6 hours. Mixer 1 requires 4.5 hours to produce a batch of product 1 and mixer 2 requires 5 hours to produce a batch of product 2. The reactor has a maximum capacity of 150 kg and can therefore not fulfil the raw material requirements for both mixer 1 and 2 with a single batch. The amount of water required to clean mixer

Table 8.1 Raw material requirements for the first illustrative example

	Product 1	Product 2
Raw material from reactor (kg)	100	150
Other raw material besides water (kg)	300	100
Water as raw material (kg)	1600	1750
Total (kg)	2000	2000

one is 300 kg and mixer two 400 kg. The reactor requires no cleaning. The duration of a washout in either mixer is 30 minutes.

Each wastewater storage vessel has a maximum storage capacity of 3000 kg and with the wastewater having a maximum residence time of 6 h in each storage vessel. Storing water for longer than 6 h could compromise the integrity of the wastewater and therefore pose a possible risk to the product.

The required production over a 36 h time horizon is 2 batches of product 1 and 3 batches of product 2. The objective function used for this example is the minimisation of effluent, since the total production is given.

It is important to note the impact of the reactor on the resulting model. Since the reactor does not use water at all, water mass balances around the reactor are not required. The reactor can also be excluded from the reuse scheduling as the operation of the reactor does not directly affect the reuse of water. However, task scheduling constraints are still required for the reactor as are raw material and product mass balances.

The resulting model was formulated in GAMS 22.0 and solved using CPLEX 9.1.2. The solution was found in 0.078 CPU seconds on a Pentium 4, 3.2 GHz processor. The optimal number of time points was 11. The resulting model had 132 binary variables. The resulting schedule is shown in Fig. 8.3. The numbers given in Fig. 8.2 give the amount of water reused in product. The resulting schedule only produces 700 kg of effluent, at the end of the time horizon. This results in a 61% reduction in the amount of water when compared to the same schedule without reuse of water in product.

As one can notice from Fig. 8.3 mixer 1 and mixer 2 make use of a storage vessel throughout the time horizon. Mixer 1 sends 300 kg to storage vessel 1 and later receives 300 kg from the same storage vessel. Mixer 2 sends and receives water from storage vessel 2 twice in the time horizon, respectively. A further observation is that no water is stored longer than 7 h, in line with the maximum storage time given in the problem.

It is important to note that there is no effluent produced during the time horizon, other than at the end of the last batch of each product. The wastewater that is generated after a mixer is cleaned after the last batch of a product has to be discarded. The wastewater cannot be stored since it is not known when the wastewater will be used. In general this is the case with any such problem, since the reuse of water can only occur if a subsequent batch of a compatible product has to be produced.

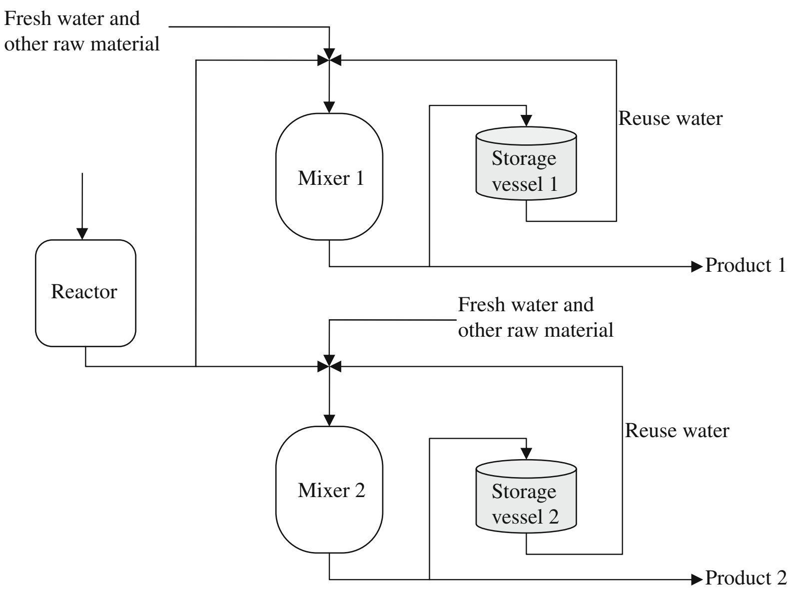


Fig. 8.2 Process diagram for first illustrative example operating in zero effluent mode

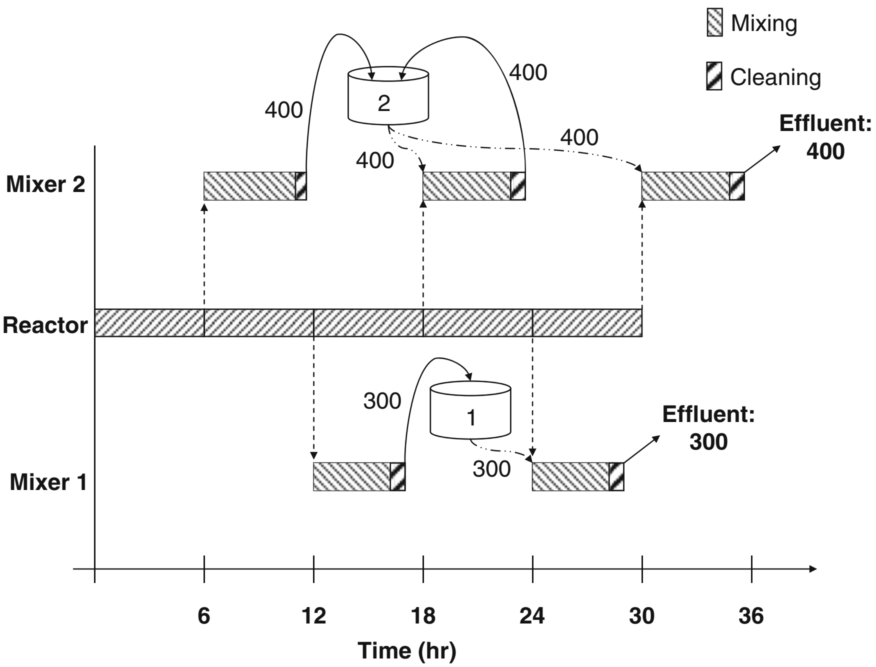


Fig. 8.3 Resulting schedule for illustrative example one using the zero effluent scheduling model

8.5.2 Second Illustrative Example

The synthesis illustrative example involves the design of a small mixing operation in which three products are produced. The objective is to determine the optimal number of processing units and the size of storage vessel needed to meet the required production and produce near zero effluent. The size of the processing units has been set to 2000 kg, due to packaging requirements. Wastewater is generated from the cleaning of the processing units after product has been removed from the unit. The wastewater generated from the cleaning operation is contaminated with the residue of the product mixed in the processing unit and can only be reused in a batch of the same product as the contaminant. Multiple batches of a product are not restricted to the same processing unit, therefore, the resulting processing units can mix any product.

The average production required in a 24 h period is 2 batches of product 1, 3 batches of product 2 and 3 batches of product 3. The composition of each product is given in Table 8.2 and the average processing time of each product is also given. The duration of a washout in each mixer is 30 min.

The maximum number of storage vessels was one in this example, with the maximum capacity of the storage vessel being 2000 kg and the minimum operating capacity being 100 kg. To ensure that different types of wastewater were not mixed in the storage vessel, constraint (8.63) was included in the model. Constraint (8.63) states that the time at which water containing a certain contaminant enters a storage vessel must be after the time at which water containing a different contaminant entered and exited the storage vessel.

$$tzs_{\text{in}}(s_{\text{in}}, j, p) \geq \sum_{s_{\text{in}}', j, p'} tzs_{\text{out}}\left(s_{\text{in}}', j, p'\right) - \sum_{s_{\text{in}}', j, p'} tzs_{\text{in}}\left(s_{\text{in}}', j, p'\right),$$
$$\forall\, j \in J, s_{\text{in}}, s_{\text{in}}' \in S_{\text{in}}, p, p' \in P, p > p' \tag{8.63}$$

Apart from the storage vessel, the maximum number of processing units that can be used is 4. The amount of water used to clean each processing unit was not the same, due to the fact that the processing units are not assumed to be identical. The amount of water used to clean processing unit 1 is 500 kg, processing unit 2 is 600 kg, processing unit 3 is 700 kg and processing unit 4 is 400 kg.

Table 8.2 Product compositions and processing times for the second illustrative example

Product	Amount of fresh water (kg)	Amount of other raw material (kg)	Processing time (h)
Product 1	1600	400	8
Product 2	1650	350	5
Product 3	1800	200	7

Table 8.3 Cost data for the second illustrative example

Constant	Value
α_{unit}	12 737 (c.u./unit)
α_{stor}	5617 (c.u.)
β_{stor}	0.25 (c.u./t)
C_e	7 (c.u./kg effluent)

The cost data used for the example is given in Table 8.3. The β_{unit} value for the unit found in the objective given in constraint (8.62) is zero in this example. The cost data given in Table 8.3 is randomly chosen data.

The resulting model for the example was formulated in GAMS 22.0 and solved using the CPLEX 9.1.2 solver. The resulting model was linear due to the fact that the contaminant mass in the wastewater was assumed negligible. The solution was found in 97.5 CPU seconds using the same processor as the previous example. The model had 421 binary variables and required 8 time points for the optimal solution. The resulting process required no storage vessel for wastewater and 3 mixing vessels. The optimal value of the objective function was 48 152 c.u. The resulting schedule is shown in Fig. 8.4. The effluent generated using the resulting schedule is 1300 kg, which is a 68% reduction in effluent compared to the same plant not operating in zero effluent mode.

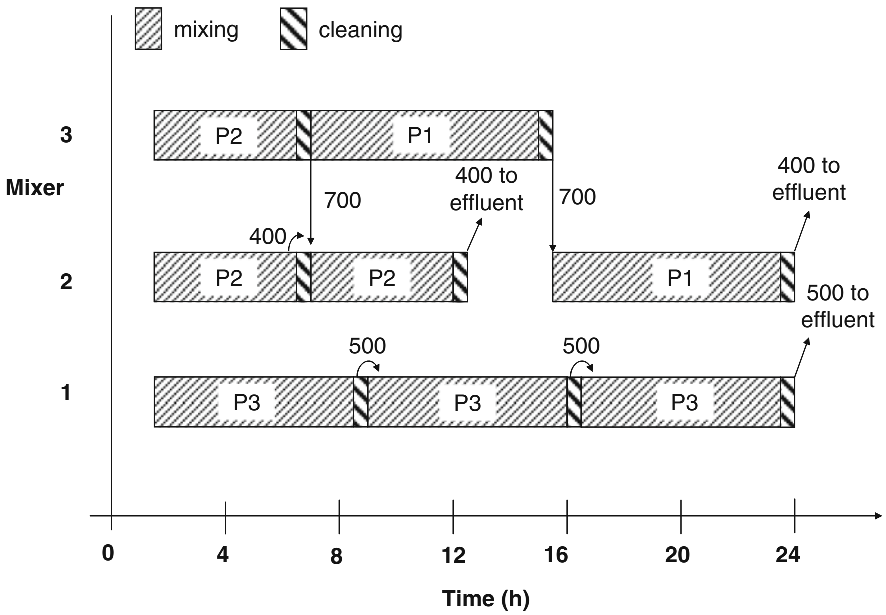


Fig. 8.4 Resulting schedule for the second illustrative example

In Fig. 8.4 the product produced by each mixer is represented by the letter “P” and then the corresponding product number. The values given in the figure represent the amount of water reused between the various mixing vessels.

As can be seen in Fig. 8.4 mixer 1 mixes product 3 exclusively and recycles the resulting effluent directly for two of the three batches of product 3. At the end of the third batch of product 3, 500 kg of water is discarded as effluent since there are no further reuse possibilities. Mixers 2 and 3 each produce products 1 and 2. Mixer 2 produces 2 batches of product 2 and after the second batch, 400 kg of water is discarded as effluent. Mixer 2 also mixes one batch of product 1, which after the completion of the cleaning operation produces 400 kg of effluent. Mixer 3 does not produce any effluent. The reason there is effluent generation is that after the final batch of product is mixed there is no further opportunity for reuse, the effluent must then be discarded.

8.6 Conclusions

The methodology presented above deals with the scheduling and synthesis of operations operating in a near zero effluent fashion. The effluent is reduced by reusing wastewater as part of product formulation. Reuse of water in this manner allows, under the correct conditions, the generation of near zero effluent.

Two formulations were derived. The first deals with minimising the amount of effluent produced from an operation where wastewater can be reused in product formulation and the plant structure is known. The minimisation is achieved by scheduling the operation in such a manner as to maximise the opportunity for wastewater reuse. The second deals with the synthesis of a batch process operating in zero effluent mode. The formulation determines the number and size of processing and storage vessels as to minimise the cost of the equipment and the amount of effluent produced from the resulting operation, while achieving the required production.

The first zero effluent scheduling formulation, was applied to an example, where the wastewater was reduced by 61%. In the resulting schedule effluent was only produced at the end of the time horizon or when the required production was achieved and there were no further opportunities for wastewater reuse.

The zero effluent synthesis formulation was applied to a second illustrative example. In the example the number of processing units and the size of the central storage vessel were not known. The resulting plant required only 3 processing units and no storage vessel. The resulting schedule produced 68% less effluent than the same operation without wastewater reuse.

The formulations derived for the zero effluent mode of operation suffer from similar problems as the previous models, in that the resulting models can reach large proportions when the number of time points is large or the problem involves a large number of processing units, storage vessels and states. The size of the problem might result in problems becoming intractable.

The large size of a resulting model is due to the fact that the number of variables, including the binary variables, is directly dependent on the number of time points. The larger the number of time points, the larger the number of variables. This affect is most remarked in the number of binary variables, where the number of binary variables has a direct affect on the solution time, i.e., the larger the number of binary variables the longer the solution times.

The binary variable dimension is further increased in the above formulations due to the fact that the binary variables describing direct and indirect reuse all carry state information, i.e., information on the state present in the water. For example, in the previous methodologies, the multiple contaminant and multiple storage vessels, there are $j \times j \times p$ binary variables for direct recycle/reuse. However, in the zero effluent methodology there are $s_{\text{in}} \times j \times j \times p$ binary variables for direct recycle/reuse.

8.7 Exercise

Task: Apply the presented mathematical formulation to the first and second illustrative examples and verify the results presented in Figs. 8.3 and 8.4.

References

Glover, F., 1975. Improved linear integer programming formulation of nonlinear problems. Manag. Sci., 22(4): 455–460

Gouws, J., Majozi, T., 2008. Impact of multiple storage in wastewater minimisation for multi-contaminant batch plants: towards zero effluent. Ind. Eng. Chem. Res., 47: 369–379

Chapter 9
Wastewater Minimisation Using Inherent Storage

Overview In this chapter, the PIS philosophy that is presented in detail in Chapter 3 of this textbook is applied within the context of wastewater minimisation. The overall idea lies in the exploitation of idleness which is an inherent feature in batch plants. Using processing units that would otherwise be idle for wastewater storage invariably reduces the need for dedicated reusable water storage facility, thereby mitigating the spatial requirements for the entire process. The first section of this chapter provides some background to the methodology. The second section deals with the exact problem considered in the methodology. The mathematical formulation is presented in the third section. This section is followed by a section on the solution procedures adopted. The fifth section provides two illustrative examples demonstrating the application of the methodology. The conclusions are found in the last section.

9.1 Inherent Storage Background

Batch processing is generally undertaken within a limited space. Therefore, space for storage vessels is often problematic. Intermediate storage for wastewater is intrinsic in wastewater minimisation since it allows for the bypassing of time restrictions on reuse. It thus allows for additional wastewater reuse opportunities over and above direct recycle/reuse opportunities. Due to the space restrictions, one would like to minimise the size of the storage for wastewater, whilst not impacting negatively on the opportunities to reduce effluent.

Furthermore, in any batch processing facility there are idle processing units at some stage within the time horizon of interest. These units are generally those that are not the bottleneck in the product recipe. The idle processing units constitute a loss in returns on capital investment. Any processing unit is, in essence, a storage vessel. Therefore, it is possible, under the correct conditions, to use the idle processing units as storage vessels for wastewater. In doing this one is able to increase the utilisation of the processing units, thus increasing the return on investment. In doing this one can also reduce the size of the central storage vessel.

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_9,

The method presented below utilises inherent storage in idle processing units within a wastewater minimisation framework to allow for the minimisation of effluent and storage. The methodology can be used to either minimise the size of the central storage vessel or provide alternative storage opportunities in the wastewater minimisation problem.

The methodology is derived for wastewater contaminated with only single contaminants. As with the previous methodologies, this methodology is based on the scheduling framework proposed by Majozi and Zhu (2001). Furthermore, the methodology is based on the uneven discretization of the time horizon. The two problems considered are formally stated in the section following this one.

9.2 Problem Statement

The two problems considered in the derivation of the methodology are formally given below. In essence the problems share many of the same characteristics, but the objectives are slightly different.
Stated formally the first problem is as follows.
Given the following data,

(i) the maximum inlet and outlet concentration for each process in the plant,
(ii) the mass load in each operation,
(iii) the average task duration of each operation,
(iv) the number of processing units and the capacity of each, and
(v) the time horizon of interest,

determine the minimum size of the central storage vessel that is concomitant with the minimum wastewater generation through the exploitation of inherent storage possibilities in idle processing units.

The second problem shares some characteristics of the first problem and is stated as follows.
Given the following data,

(i) the maximum inlet and outlet concentration for each process in the plant,
(ii) the mass load in each operation,
(iii) the average task duration of each operation,
(iv) the number of processing units and the capacity of each,
(v) the capacity of the central storage vessel, and
(vi) the time horizon of interest,

determine the process schedule that will result in the minimum wastewater generation through the exploitation of all available reuse opportunities. The reuse opportunities available are either in the form of direct reuse, from processing unit to processing unit, or indirect reuse through the central storage vessel or inherent storage in idle processing units.

The mathematical formulation that addresses the above two problems share the same constraints, the only difference is in the solution procedure and the objective functions.

9.3 Mathematical Formulation for Inherent Storage

The mathematical formulation for exploitation of inherent storage comprises of the following sets, variables and parameters.

Sets

$$
\begin{aligned}
P &= \{p|p = \text{time point}\}\\
J &= \{j|j = \text{unit}\}\\
C &= \{c|c = \text{contaminant}\}\\
U &= \{u|u = \text{reusable water storage vessel}\}\\
S_{\text{in}} &= \{S_{\text{in}}|S_{\text{in}} = \text{input state into any unit}\}\\
S_{\text{out}} &= \{S_{\text{out}}|S_{\text{out}} = \text{output state from any unit}\}\\
S_{\text{in},j} &= \left\{S_{\text{in},j}|S_{\text{in},j} = \text{input state into unit } j\right\} \subseteq S_{\text{in}}\\
s_{\text{out},j} &= \left\{s_{\text{out},j}|s_{\text{out},j} = \text{output state from unit } j\right\} \subseteq S_{\text{out}}\\
S &= \{s|s \text{ is any state}\} = S_{\text{in}} \cup S_{\text{out}}
\end{aligned}
$$

The mathematical formulation comprises of a number of mass balance constraints and scheduling constraints. The derivation of each of the constraints in the formulation is done considering a superstructure depicting the processes concerned. The sets, variables and parameters used in this chapter are all described in the nomenclature list.

9.3.1 Superstructure Used in the Mathematical Formulation

The superstructure on which the mathematical formulation is based is given in Fig. 9.1. Figure 9.1a shows a water using unit and Fig. 9.1b shows the overall plant superstructure. Figure 9.1a one would notice that water entering a unit is comprised of freshwater, water from inherent storage, direct recycle/reuse water and water from the central storage vessel. This is similar for water leaving a unit. Figure 9.1b shows that each water using operation can also operate as a storage vessel for wastewater within the overall plant superstructure.

Mass balance constraints are derived around each processing unit and the central storage vessel. The mass balance constraints around a processing unit are presented

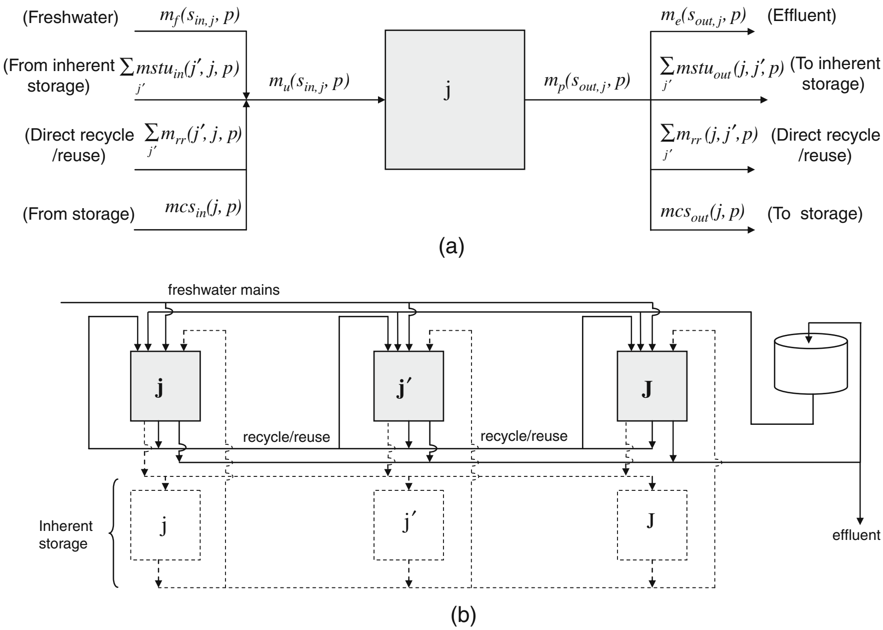


Fig. 9.1 Superstructure for the inherent storage model with a central storage vessel (Gouws and Majozi, 2009)

for both the case where the unit is processing raw material and where a processing unit is operating as an inherent storage vessel.

9.3.2 Mass Balance Constraints

Mass Balances Around a Unit Processing Raw Material

The mass balances considered around a processing unit processing raw material comprise of water and contaminant balances. An inlet water balance is given in constraint (9.1). This constraint states that the water entering a processing unit comprises of freshwater, water from the central storage vessel, directly recycle/reused water and water from inherent storage. A similar balance is presented in constraint (9.2) for water leaving a processing unit. In constraint (9.2), water leaving a processing unit is either discarded as effluent, sent to the central storage vessel for reuse at a later time, directly recycled/reused or sent to another processing unit operating as a storage vessel. It is assumed that the task in the processing unit does not generate additional water or consume water. Consequently, the amount of water exiting a processing unit is equal to the amount of water entering the processing unit, given in constraint (9.3). In the case where the operation within a processing unit does generate or consume water, additional terms must be included in constraint (9.3) to account for the water gain or loss, respectively.

$$m_u\left(s_{\text{in},j}, p\right) = m_f\left(s_{\text{in},j}, p\right) + mcs_{\text{out}}(j, p) + \sum_{j'} m_{rr}\left(j', j, p\right) + \sum_{j'} mstu_{\text{out}}\left(j', j, p\right), \quad \forall j, j' \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.1}$$

$$m_p\left(s_{\text{out},j}, p\right) = m_e\left(s_{\text{out},j}, p\right) + mcs_{\text{in}}(j, p) + \sum_{j'} m_{rr}\left(j, j'p\right) + \sum_{j'} mstu_{\text{in}}\left(j, j'p\right), \quad \forall j, j' \in J,\ s_{\text{out},j} \in S_{\text{out},j},\ p \in P \tag{9.2}$$

$$m_u\left(s_{\text{in},j}, p-1\right) = m_p\left(s_{\text{out},j}, p\right), \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j}, s_{\text{out},j} \in S_{\text{out},j},\ p \in P, p > p_1 \tag{9.3}$$

An inlet contaminant balance is given in constraint (9.4). This constraint states that the contaminant mass entering a processing unit comprises of contaminant mass in water that is directly recycled/reused, the contaminant mass in water from inherent storage and the contaminant mass in water from central storage. A contaminant balance over a processing unit is given in constraint (9.5). This constraint states that contaminant mass leaving a processing unit is comprised of contaminant mass that entered the unit and mass load added due to the task in the unit.

$$\begin{aligned} m_u\left(s_{\text{in},j}, p\right) ci_{\text{in}}\left(s_{\text{in},j}, p\right) = & \sum_{j'} m_r\left(j', j, p\right) ci_{\text{out}}(s_{\text{out},j'}, p) \\ & + \sum_{j'} mstu_{\text{out}}\left(j' j, p\right) cstu_{\text{out}}(j', p) + mcs_{\text{out}}(j, p)\, cs_{\text{out}}(p), \\ & \forall j, j' \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ s_{\text{out},j'} \in S_{\text{out},j},\ p \in P \end{aligned} \tag{9.4}$$

$$\begin{aligned} m_p\left(s_{\text{out},j}, p\right) ci_{\text{out}}\left(s_{\text{out},j}, p\right) = & m_u\left(s_{\text{in},j}, p-1\right) ci_{\text{in}}\left(s_{\text{in},j}, p-1\right) \\ & + M(s_{\text{in},j}) y\left(s_{\text{in},j}, p-1\right), \\ & \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ s_{\text{out},j} \in S_{\text{out},j}, p \in P,\ p > p_1 \end{aligned} \tag{9.5}$$

The maximum inlet and outlet concentration of water entering a processing unit is ensured through constraints (9.6) and (9.7). Due to the fact that the system has only a single contaminant present, the outlet concentration of each processing unit is set to the respective maximum. This condition must hold for the resulting wastewater reuse network to be optimal (Savelski and Bagajewicz, 2000). The amount of water used for processing and the amount recycled/reused is limited to the maximum amount allowable in constraints (9.8) and (9.9). It must be noted that if the amount of water required by each operation in fixed, then constraints (9.7) and (9.8) must be reformulated. In this case the outlet concentration is not necessarily at its maximum. Hence constraint (9.7) is reformulated to ensure that the outlet concentration does not exceed the maximum. The operator in constraint (9.8), in this case, is replaced with an equal sign to ensure the fixed amount of water condition holds.

$$ci_{\text{in}}\left(s_{\text{in},j}, p\right) \le CI_{\text{in}}^{\max}(s_{\text{in},j}) y(s_{\text{in},j}, p),\ \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.6}$$

$$ci_{\text{out}}(s_{\text{out},j}, p) = CI_{\text{out}}^{\max}(s_{\text{out},j}) y\left(s_{\text{in},j}, p-1\right), \quad \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ s_{\text{out},j} \in S_{\text{out},j},\ p \in P,\ p > p_1 \tag{9.7}$$

$$m_u\left(s_{\text{in},j}, p\right) \leq M_u^{\max}\left(s_{\text{in},j}\right) y\left(s_{\text{in},j}, p\right), \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.8}$$

$$m_{rr}\left(j', j, p\right) \leq M_u^{\max}\left(s_{\text{in},j}\right) y_r(j', j, p), \forall j, j' \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.9}$$

The maximum amount of water used in constraints (9.8) and (9.9) is calculated using constraint (9.10). Constraint (9.10) defines the maximum amount of water that can be used for a task in a processing unit while still removing the required mass load and obeying the limiting inlet and outlet concentrations. The maximum amount of water is the ratio of the mass load added to the water by the task in the processing unit to the difference between the maximum inlet and outlet concentrations.

$$M_u^{\max}\left(s_{\text{in},j}\right) = \frac{M\left(s_{\text{in},j}\right)}{CI_{\text{out}}^{\max}\left(s_{\text{out},j}\right) - CI_{\text{in}}^{\max}\left(s_{\text{in},j}\right)}, \quad \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ s_{\text{out},j} \in S_{\text{out},j} \tag{9.10}$$

Mass Balance Constraints for a Processing Unit Operating in Inherent Storage Mode

The mass balances around a processing unit operating in inherent storage mode are similar to the constraints around the central storage vessel. Constraint (9.11) is a water balance around a processing unit. This constraint states that the amount of water stored in a processing unit at any time point is the amount from the previous time point and the difference between the water entering and exiting the unit. Constraint (9.12) is the form of constraint (9.11) for the first time point. At the first time point, which corresponds to the beginning of the time horizon, the amount of water stored in a processing unit is zero. Constraint (9.13) ensures that the amount of water stored in a processing unit is less than the capacity of the unit. This constraint further ensures that if the processing unit is undertaking a task, then no water can be stored in the unit.

$$qu(j, p) = qu(j, p-1) + \sum_{j'} mstu_{\text{in}}\left(j', j, p\right) - \sum_{j} mstu_{\text{out}}\left(j, j', p\right), \quad \forall j \in J, p \in P,\ p > p_1 \tag{9.11}$$

$$qu(j, p_1) = 0, \forall j \in J \tag{9.12}$$

$$qu(j, p) \leq Q_u^{\max}\left(1 - y\left(s_{\text{in},j}, p\right)\right), \forall, j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.13}$$

Constraint (9.14) is a contaminant balance over a processing unit operating in inherent storage mode. It is assumed in constraint (9.14) that the unit is ideally mixed.

$$cstu_{\text{out}}(j,p) = \frac{qu(j,p-1)\,cstu_{\text{out}}(j,p-1) + \sum_{j'} mstu_{\text{in}}(j',j,p)\,ci_{\text{out}}(j',p)}{qu(j,p-1) + \sum_{j'} mstu_{\text{in}}(j',j,p)},$$
$$\forall j,j' \in J, p \in P, p > p_1 \qquad (9.14)$$

The amount of water entering a processing unit for storage must be less than the capacity of the unit, which is ensured through constraint (9.15). Furthermore, the amount of water used from inherent storage must be less than the maximum amount required in the sink task, given in constraint (9.16).

$$mstu_{\text{in}}(j',j,p) \le Q_u^{\max} ystu_{\text{in}}(j',j,p),\ \forall j,j' \in J, p \in P \qquad (9.15)$$

$$mstu_{\text{out}}(j,j',p) \le M_u^{\max}\left(s_{\text{in},j'}\right) ystu_{\text{out}}(j,j',p),\ \forall j,j' \in J,\ s_{\text{in},j'} \in S_{\text{in},j},\ p \in P \qquad (9.16)$$

Central Storage Mass Balances

The mass balance constraints around the central storage vessel are given in constraints (9.17), (9.18), (9.19), (9.20), (9.21), (9.22), (9.23) and (9.24). These constraints are very similar to those presented by Majozi (2005) and will therefore not be discussed in great detail.

Constraints (9.17) and (9.18) are water balances around the central storage vessel, while constraints (9.19), (9.20) and (9.21) are contaminant balances around the storage vessel. Constraint (9.19) is an inlet contaminant balance, whilst constraint (9.20) is a contaminant balance over the storage vessel and constraint (9.21) defines the initial concentration in the storage vessel. Constraints (9.22) and (9.23) ensure that the amount of water stored in the storage vessel and the amount of water entering the storage vessel is less than the capacity of the storage vessel. Constraint (9.24) ensures that the amount of water reused from the central storage vessel to a processing unit is less than the maximum amount of water required for that specific task in the processing unit.

$$qs(p) = qs(p-1) + \sum_j mcs_{\text{in}}(j,p) - \sum_j mcs_{\text{out}}(j,p),$$
$$\forall j \in J, p \in P, p > p_1 \qquad (9.17)$$

$$qs(p_1) = Q_s^0 - \sum_j mcs_{\text{out}}(j,p_1),\ \ \forall j \in J \qquad (9.18)$$

$$cs_{\text{in}}(p) = \frac{\sum_j \left(mcs_{\text{in}}(j,p)\,ci_{\text{out}}(s_{\text{out},j},p)\right)}{\sum_j mcs_{\text{in}}(j,p)},\ \forall j,j' \in J, s_{\text{out},j} \in S_{\text{out},j}, p \in P \qquad (9.19)$$

$$cs_{\text{out}}(p) = \frac{qs(p-1)\, cs_{\text{out}}(p-1) + \left(\sum_j mcs_{\text{in}}(j,p)\right) cs_{\text{in}}(p)}{qs(p-1) + \sum_j mcs_{\text{in}}(j,p)}, \quad \forall j, j' \in J, p \in P, p > p_1 \tag{9.20}$$

$$cs_{\text{out}}(p_1) = CS^0_{\text{out}} \tag{9.21}$$

$$qs(p) \le Q_s^{\max}, \ \forall\, p \in P \tag{9.22}$$

$$mcs_{\text{in}}(j,p) \le Q_s^{\max} ys_{\text{in}}(j,p), \quad \forall j \in J, p \in P \tag{9.23}$$

$$mcs_{\text{out}}(j,p) \le M_u^{\max}\left(s_{\text{in},j}\right) ycs_{\text{out}}(j,p), \quad \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.24}$$

The above constraints deal with the mass flows between the various units in a batch plant. They do not consider the timing of the streams, tasks and such. Therefore, further constraints have to be derived to ensure the correct sequencing and scheduling of the processes, streams and tasks.

9.3.3 *Sequencing and Scheduling Constraints*

The constraints that deal with the scheduling aspect of the processes involved can be divided into five groups, with each group dealing with a distinct scheduling aspect. The first group comprises of constraints pertinent to task scheduling. The constraints in the second group are centred on the scheduling of the direct recycle/reuse. The third group of constraints deal with inherent storage scheduling and the fourth group with the scheduling of the central storage vessel. The final group deals with feasibility constraints and time horizon constraints.

Task Scheduling Constraints

The task scheduling constraints ensure the correct scheduling of tasks within an operation. Constraint (9.25) ensures that a task in a processing unit is complete before the next task begins. Constraints (9.26) and (9.27) ensure that a task starting or finishing in a processing unit at a later time point do so at a later absolute time within the time horizon. A task duration constraint is presented in constraint (9.28), which states that the finishing time of a task occurs at a time equal to the starting time of the task and the task duration.

$$t_u\left(s_{\text{in},j}, p\right) \ge t_p\left(s_{\text{out},j}, p\right) - H\left(2 - y\left(s_{\text{in},j}, p\right) - y\left(s_{\text{in},j}', p-1\right)\right), \quad \forall j \in J,\ s_{\text{in},j}', s_{\text{in},j} \in S_{\text{in},j},\ s_{\text{out},j} \in S_{\text{out},j},\ p \in P,\ p > p_1 \tag{9.25}$$

$$t_u\left(s_{\text{in},j}, p\right) \ge t_u\left(s_{\text{in},j}', p'\right) - H\left(2 - y\left(s_{\text{in},j}, p\right) - y\left(s_{\text{in},j}', p'\right)\right), \quad \forall j \in J,\ s_{\text{in},j}, s_{\text{in},j}' \in S_{\text{in},j},\ p, p' \in P,\ p \ge p' \tag{9.26}$$

$$t_p\left(s_{\text{out},j},p\right) \geq t_p\left(s_{\text{out},j},p'\right) - H\left(2 - y\left(s_{\text{out},j},p\right) - y\left(s_{\text{out},j}{}',p'\right)\right), \quad \forall j \in J, s_{\text{out},j}, s_{\text{out},j}{}' \in s_{\text{out},j}, P, \, p' \in P, \, p \geq p' \tag{9.27}$$

$$t_p\left(s_{\text{out},j},p\right) = t_u\left(s_{\text{in},j}, \, p-1\right) + \tau\left(s_{\text{in}}\right) y\left(s_{\text{in},j}, \, p-1\right), \quad \forall j \in J, \, s_{\text{in},j} \in S_{\text{in},j}, \, s_{\text{out},j} \in s_{\text{out},j}, s_{\text{in}} \in S_{\text{in}}, \, p \in P, \, p > p_1 \tag{9.28}$$

Following the task scheduling constraints are the constraints dealing with the scheduling of the direct recycle/reuse of wastewater.

Recycle/Reuse Sequencing Constraints

The direct recycle/reuse scheduling constraints are given in constraints (9.29), (9.30), (9.31), (9.32) and (9.33). Constraint (9.29) states that water can only be recycled/reused to a processing unit provided the unit is operating at that time point. However, the constraint also states that the recycle/reuse of water is not a prerequisite for the operation of a processing unit. Constraints (9.30) and (9.31) ensure that the time at which water is recycled/reused corresponds to the time at which the water is produced. Constraints (9.32) and (9.33) ensure that the time at which water is recycled/reused corresponds to the starting time of the task using the water.

$$y_{rr}\left(j,j',\, p\right) \leq \sum_{s_{\text{in}}} y\left(s_{\text{in},j}{}',p\right), \;\; \forall j,j' \in J, \; s_{\text{in},j} \in S_{\text{in},j} p \in P \tag{9.29}$$

$$t_{rr}\left(j,j',\, p\right) \leq t_p\left(s_{\text{out},j},p\right) + H\left(2 - y_{rr}\left(j,j',\, p\right) - y\left(s_{\text{in},j}, \, p-1\right)\right), \quad \forall \, j,j' \in J, \; s_{\text{in},j} \in S_{\text{in},j}, \; s_{\text{out},j} \in s_{\text{out},j}, \; p \in P, \; p > p_1 \tag{9.30}$$

$$t_{rr}\left(j,j',\, p\right) \geq t_p\left(s_{\text{out},j},p\right) - H\left(2 - y_{rr}\left(j,j',\, p\right) - y\left(s_{\text{in},j}, \, p-1\right)\right), \quad \forall \, j,j' \in J, \; s_{\text{in},j} \in S_{\text{in},j}, \; s_{\text{out},j} \in s_{\text{out},j}, \; p \in P, \; p > p_1 \tag{9.31}$$

$$t_{rr}\left(j,j',\, p\right) \leq t_u\left(s_{\text{in},j}{}',p\right) + H\left(2 - y_{rr}\left(j,j',\, p\right) - y\left(s_{\text{in},j}{}',p\right)\right), \quad \forall j,j' \in J, \; s_{\text{in},j}{}' \in S_{\text{in},j}{}', p \in P \tag{9.32}$$

$$t_{rr}\left(j,j',\, p\right) \geq t_u\left(s_{\text{in},j}{}',p\right) - H\left(2 - y_{rr}\left(j,j',\, p\right) - y\left(s_{\text{in},j}{}',p\right)\right), \quad \forall j,j' \in J, \; s_{\text{in},j}{}' \in S_{\text{in},j}{}', p \in P \tag{9.33}$$

Apart from direct reuse scheduling, constraints also have to be derived to ensure the correct scheduling of indirect reuse through inherent storage.

Inherent Storage Scheduling Through Inherent Storage

In published mathematical formulations scheduling constraints for wastewater storage in processing units are not required, since this option is ignored. However, in this formulation the option to store wastewater in idle processing units is included, hence scheduling constraints that govern this occurrence are required.

Constraint (9.34) states that if water is being sent to a processing unit for storage at time point p, the source processing unit must have operated at the previous time point. The constraint also ensures that a processing unit can operate without sending water to another processing unit for storage. A similar constraint is given in constraint (9.35). Constraint (9.35) states that water can only be sent to a processing unit from inherent storage if the receiving unit is operating at that time point. As with the previous constraint, it is not a prerequisite for a processing unit to receive water from inherent storage in order to operate.

$$ystu_{\text{in}}\left(j,j',p\right) \leq \sum_{s_{\text{in}}} y\left(s_{\text{in},j},p-1\right), \quad \forall j,j' \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P,\ p > p_1 \tag{9.34}$$

$$ystu_{\text{out}}\left(j',j,p\right) \leq \sum_{s_{\text{in}}} y\left(s_{\text{in},j},p\right), \quad \forall j,j' \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.35}$$

Constraint (9.36) ensures that a unit does not start processing material and receive wastewater for storage at the same time point.

$$ystu_{\text{in}}\left(j',j,p\right) + \sum_{s_{\text{in}}} y\left(s_{\text{in},j},p\right) \leq 1, \quad \forall j,j' \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.36}$$

Apart from the binary variable constraints given above, scheduling constraints are also required to ensure the transfer of water occurs at the correct time in the time horizon.

Constraints (9.37) and (9.38) ensure that the time at which water is transferred from a processing unit to another processing unit for storage occurs at the same time at which the wastewater is produced. Similarly, constraints (9.39) and (9.40) ensure that the time at which water from inherent storage is transferred to a processing unit for usage occurs at the same time at which the task commences in the receiving unit.

$$tstu_{\text{in}}\left(j,j',p\right) \geq t_p\left(s_{\text{out},j},p\right) - H\left(2 - ystu_{\text{in}}\left(j,j',p\right) - y\left(s_{\text{in},j},p-1\right)\right), \quad \forall j,j' \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ s_{\text{out},j} \in s_{\text{out},j},\ p \in P,\ p > p_1 \tag{9.37}$$

$$tstu_{\text{in}}\left(j,j',p\right) \leq t_p\left(s_{\text{out},j},p\right) + H\left(2 - ystu_{\text{in}}\left(j,j',p\right) - y\left(s_{\text{in},j},p-1\right)\right), \quad \forall j,j' \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ s_{\text{out},j} \in S_{\text{out},j},\ p \in P,\ p > p_1 \tag{9.38}$$

$$tstu_{\text{out}}\left(j',j,p\right) \geq t_u\left(s_{\text{in},j},p\right) - H\left(2 - ystu_{\text{out}}\left(j',j,p\right) - y\left(s_{\text{in},j},p\right)\right), \quad \forall j,j' \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.39}$$

$$tstu_{\text{out}}\left(j',j,p\right) \leq t_u\left(s_{\text{in},j},p\right) + H\left(2 - ystu_{\text{out}}\left(j',j,p\right) - y\left(s_{\text{in},j},p\right)\right), \quad \forall j,j' \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.40}$$

Constraint (9.41) ensures that the time at which water is transferred to a processing unit for storage occurs after a task has finished in the receiving processing unit. Constraint (9.42) ensures the time at which water is transferred to a processing unit for storage is before the receiving processing unit starts processing the next task. The time at which water is transferred from a processing unit to another processing unit for usage must occur before the starting time of the next task in the source processing unit. This is ensured through constraint (9.43).

$$tstu_{\text{in}}\left(j',j,p'\right) \geq t_p\left(s_{\text{out},j},p\right) - H\left(2 - ystu_{\text{in}}\left(j',j,p'\right) - y\left(s_{\text{in},j},p\right)\right), \quad \forall j,j' \in J, p \in P,\ p' \geq p \tag{9.41}$$

$$tstu_{\text{in}}\left(j',j,p'\right) \leq t_u\left(s_{\text{in},j},p\right) + H\left(2 - ystu_{\text{in}}\left(j',j,p'\right) - y\left(s_{\text{in},j},p\right)\right), \quad \forall j,j' \in J, p \in P, p' \leq p \tag{9.42}$$

$$tstu_{\text{out}}\left(j,j',p'\right) \leq t_u\left(s_{\text{in},j},\, p\right) - H\left(2 - ystu_{\text{out}}\left(j,j',p'\right) - y\left(s_{\text{in},j},\, p\right)\right), \quad \forall j,j' \in J,\, s_{\text{in},\,j} \in S_{\text{in},j},\, p \in P,\, p' \leq p \tag{9.43}$$

As with any water storage vessel, water stored can be sent to multiple sinks and the processing unit can receive water from multiple sources. The scheduling of these multiple streams entering and exiting the unit has to be considered.

Constraint (9.44) ensures that water exiting a processing unit operating in inherent storage mode at a later time point does so at a later time in the time horizon. Constraints (9.45) and (9.46) ensure if two water streams are leaving a processing unit at a time point, the time at which each of these occurs in the time horizon is the same

$$tstu_{\text{out}}\left(j,j'',p\right) \geq tstu_{\text{out}}\left(j,j',p'\right) - H\left(2 - ys_{out}\left(j,j'',p\right) - ys_{\text{out}}\left(j,j',p'\right)\right), \quad \forall j,j',j'' \in J,P,\, p' \in P,\, p > p' \tag{9.44}$$

$$tstu_{\text{out}}\left(j,j',\, p\right) \geq tstu_{\text{out}}\left(j,j'',p\right) - H\left(2 - ystu_{\text{out}}\left(j,j',\, p\right) - ystu_{\text{out}}\left(j,j'',p\right)\right), \quad \forall j,j',j'' \in J, p \in P \tag{9.45}$$

$$tstu_{\text{out}}\left(j,j',\, p\right) \leq tstu_{\text{out}}\left(j,j'',p\right) + H\left(2 - ystu_{\text{out}}\left(j,j',\, p\right) - ystu_{\text{out}}\left(j,j'',p\right)\right), \quad \forall j,j',j'' \in J, p \in P \tag{9.46}$$

Similar constraints also hold for multiple water streams entering a processing unit operating in inherent storage mode. Constraint (9.47) ensures that the time at which water enters a processing unit at a later time point corresponds to a later time in the time horizon. Constraints (9.48) and (9.49) ensure that if two streams enter a processing unit at a time point, the timing of each stream corresponds to the same time in the time horizon.

$$tstu_{\text{in}}\left(j,j',\, p\right) \geq tstu_{\text{in}}\left(j,j'',p'\right) - H\left(2 - ys_{\text{in}}\left(j,j',\, p\right) - ys_{\text{in}}\left(j,j'',p'\right)\right), \quad \forall j,j',j'' \in J,P,\, p' \in P,\, p > p' \tag{9.47}$$

$$tstu_{\text{in}}\left(j,'j,\, p\right) \geq tstu_{\text{in}}\left(j'',j,\, p\right) - H\left(2 - ystu_{\text{in}}\left(j',j,\, p\right) - ystu_{\text{in}}\left(j'',j,\, p\right)\right), \quad \forall j,j',j'' \in J, p \in P \tag{9.48}$$

$$tstu_{\text{in}}\left(j',j,\, p\right) \leq tstu_{\text{in}}\left(j'',j,\, p\right) + H\left(2 - ystu_{\text{in}}\left(j,'j,\, p\right) - ystu_{\text{in}}\left(j'',j,\, p\right)\right), \quad \forall j,j',j'' \in J, p \in P \tag{9.49}$$

If a stream leaves a processing unit at a later time point than when water entered the same processing unit, then the time at which this occurs must correspond to a later time in the time horizon. This is given in constraint (9.50). Furthermore, if a water stream is entering a processing unit and another water stream is leaving the processing unit at a time point, then the time at which this occurs must correspond to the same time in the time horizon. Constraints (9.51) and (9.52) ensure this.

$$tstu_{\text{out}}\left(j,j',p\right) \geq tstu_{\text{in}}\left(j'',j,p'\right) - H\left(2 - ystu_{\text{out}}\left(j,j',p\right) - ystu_{\text{in}}\left(j'',j,p'\right)\right), \\ \forall j,j',j'' \in J, P, p' \in P, p > p' \tag{9.50}$$

$$tstu_{\text{out}}\left(j,j',p\right) \geq tstu_{\text{in}}\left(j'',j,p\right) - H\left(2 - ystu_{\text{out}}\left(j,j',p\right) - ystu_{\text{in}}\left(j'',j,p\right)\right), \\ \forall j,j',j'' \in J, p \in P \tag{9.51}$$

$$tstu_{\text{out}}\left(j,j',p\right) \leq tstu_{\text{in}}\left(j'',j,p\right) + H\left(2 - ystu_{\text{out}}\left(j,j',p\right) - ystu_{\text{in}}\left(j'',j,p\right)\right), \\ \forall j,j',j'' \in J, p \in P \tag{9.52}$$

In the case where there is a central storage vessel, scheduling constraints also have to be derived to account for the timing of water entering and exiting the vessel relative to the overall operation.

Scheduling Constraints Associated with Central Storage

The constraints that ensure the correct scheduling of the central storage vessel are similar to those presented by Majozi (2005) and will therefore not be discussed in great detail.

Water can only be sent to storage provided the source process has taken place, this is ensured through constraint (9.53). Constraints (9.54) and (9.55) ensure that the time at which water is sent to central storage is the same time at which the water was produced.

$$ycs_{\text{in}}(j,p) \leq \sum_{s_{\text{in}}} y\left(s_{\text{in},j}, p-1\right), \\ \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P,\ p > p_1 \tag{9.53}$$

$$tcs_{\text{in}}(j,p) \geq t_p\left(s_{\text{out},j}, p\right) - H\left(2 - ycs_{\text{in}}(j,p) - y\left(s_{\text{in},j}, p-1\right)\right), \\ \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ s_{\text{out},j} \in S_{\text{out},j},\ p \in P,\ p > p_1 \tag{9.54}$$

$$tcs_{\text{in}}(j,p) \leq t_p\left(s_{\text{out},j}, p\right) + H\left(2 - ycs_{\text{in}}(j,p) - y\left(s_{\text{in},j}, p-1\right)\right), \\ \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ s_{\text{out},j} \in s_{\text{out},j},\ p \in P,\ p > p_1 \tag{9.55}$$

Constraint (9.56) ensures that if water is sent to a processing unit from central storage, then the processing unit is operating at that time point. Constraints (9.57) and (9.58) ensure that the time at which water is sent to a processing unit coincides with the time at which the task starts in the receiving processing unit.

$$ycs_{\text{out}}(j,p) \leq \sum_{s_{\text{in}}} y\left(s_{\text{in},j}, p\right),\ \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.56}$$

$$tcs_{\text{out}}(j,p) \geq t_u\left(s_{\text{in},j}, p\right) - H\left(2 - ycs_{\text{out}}(j,p) - y\left(s_{\text{in},j}, p\right)\right), \\ \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.57}$$

$$tcs_{\text{out}}(j,p) \leq t_u\left(s_{\text{in},j}, p\right) + H\left(2 - ycs_{\text{out}}(j,p) - y\left(s_{\text{in},j}, p\right)\right), \\ \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.58}$$

Water leaving the central storage vessel at a later time point must correspond to a later time in the time horizon, given in constraint (9.59). Constraints (9.60) and (9.61) ensure that if two streams leave the central storage vessel at a time point, then the timing of each corresponds to the same time in the time horizon.

$$tcs_{\text{out}}(j,p) \geq tcs_{\text{out}}\left(j',p'\right) - H\left(2 - ycs_{\text{out}}(j,p) - ycs_{\text{out}}\left(j',p'\right)\right), \quad \forall j,j' \in J, P, p' \in P, p \geq p' \tag{9.59}$$

$$tcs_{\text{out}}(j,p) \geq tcs_{\text{out}}\left(j',p\right) - H\left(2 - ycs_{\text{out}}(j,p) - ycs_{\text{out}}\left(j',p\right)\right), \quad \forall j,j' \in J, p \in P \tag{9.60}$$

$$tcs_{\text{out}}(j,p) \leq tcs_{\text{out}}\left(j',p\right) + H\left(2 - ycs_{\text{out}}(j,p) - ycs_{\text{out}}\left(j',p\right)\right), \quad \forall j,j' \in J, p \in P \tag{9.61}$$

Constraints (9.62), (9.63) and (9.64) are similar to constraints (9.59), (9.60) and (9.61), however, these constraints apply to multiple streams entering the central storage vessel.

$$tcs_{\text{in}}(j,p) \geq tcs_{\text{in}}\left(j',p'\right) - H\left(2 - ycs_{\text{in}}(j,p) - ycs_{\text{in}}\left(j',p'\right)\right), \quad \forall j,j' \in J, P, p' \in P, p \geq p' \tag{9.62}$$

$$tcs_{\text{in}}(j,p) \geq tcs_{\text{in}}\left(j',p\right) - H\left(2 - ycs_{\text{in}}(j,p) - ycs_{\text{in}}\left(j',p\right)\right), \quad \forall j,j' \in J, p \in P \tag{9.63}$$

$$tcs_{\text{in}}(j,p) \leq tcs_{\text{in}}\left(j',p\right) + H\left(2 - ycs_{\text{in}}(j,p) - ycs_{\text{in}}\left(j',p\right)\right), \quad \forall j,j' \in J, p \in P \tag{9.64}$$

The final scheduling constraints that deal with the scheduling of the central storage vessel are given in constraints (9.65), (9.66) and (9.67). Constraint (9.65) ensures water leaving the central storage vessel at a later time point to when water entered the vessel does so at a time corresponding to a later absolute time in the time horizon. Constraints (9.66) and (9.67) ensure that the timing of two streams entering and exiting the central storage vessel at a time point correspond to the same time in the time horizon.

$$tcs_{\text{out}}(j,p) \geq tcs_{\text{in}}\left(j',p'\right) - H\left(2 - ycs_{\text{out}}(j,p) - ycs_{\text{in}}\left(j',p'\right)\right), \quad \forall j,j' \in J, P, p' \in P, p \geq p' \tag{9.65}$$

$$tcs_{\text{out}}(j,p) \geq tcs_{\text{in}}\left(j',p\right) - H\left(2 - ycs_{\text{out}}(j,p) - ycs_{\text{in}}\left(j',p\right)\right), \quad \forall j,j' \in J, p \in P \tag{9.66}$$

$$tcs_{\text{out}}(j,p) \leq tcs_{\text{in}}\left(j',p\right) + H\left(2 - ycs_{\text{out}}(j,p) - ycs_{\text{in}}\left(j',p\right)\right), \quad \forall j,j' \in J, p \in P \tag{9.67}$$

Feasibility and Time Horizon Constraints

The final scheduling constraints are the feasibility constraints and time horizon constraints. Constraint (9.68) ensures that a processing unit can only process one task at a time. Constraint (9.69) ensures that if processing unit j is reusing water to unit j' at time point p, then unit j' cannot reuse water to unit j at the same time point.

$$\sum_{s_{\text{in},j}} y\left(s_{\text{in},j}, p\right) \leq 1, \qquad \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.68}$$

$$y_r\left(j,j', p\right) + y_r\left(j',j, p\right) \leq 1, \qquad \forall j,j' \in J, p \in P \tag{9.69}$$

The time horizon constraints are given in constraints (9.70), (9.71), (9.72), (9.73), (9.74), (9.75) and (9.76). These constraints ensure that each event within the overall operation occurs within the time horizon of interest.

$$tcs_{\text{in}}\,(j, p) \leq H, \quad \forall j \in J, p \in P \tag{9.70}$$

$$tcs_{\text{out}}\,(j, p) \leq H, \quad \forall j \in J, p \in P \tag{9.71}$$

$$t_u\left(s_{\text{in},j}, p\right) \leq H, \quad \forall j \in J,\ s_{\text{in},j} \in S_{\text{in},j},\ p \in P \tag{9.72}$$

$$t_p\left(s_{\text{out},j}, p\right) \leq H, \quad \forall j \in J, s_{\text{out},j} \in s_{\text{out},j},\ p \in P \tag{9.73}$$

$$t_{rr}\left(j,j', p\right) \leq H, \quad \forall j,j' \in J, p \in P \tag{9.74}$$

$$tstu_{\text{in}}\left(j,'j, p\right) \leq H, \quad \forall j,j' \in J, p \in P \tag{9.75}$$

$$tstu_{\text{out}}\left(j,j', p\right) \leq H, \quad \forall j,j' \in J, p \in P \tag{9.76}$$

9.3.4 Inclusion of the Maximum Outlet Concentration Condition

The application of the maximum outlet concentration condition (Savelski and Bagajewicz, 2000) allows for the simplification of constraint (9.5) and subsequent linearisation of two nonlinear terms present in the resulting constraint. This is done as follows.

Constraints (9.4) and (9.7) are substituted into constraint (9.5) resulting in constraint (9.77).

$$\begin{aligned} &CI_{\text{out}}^{\max}\left(s_{\text{out},j}\right) y\left(s_{\text{in},j}, p-1\right) m_p\left(s_{\text{out},j}, p\right) \\ &= \sum_{j'} m_{rr}\left(j',j,p-1\right) CI_{\text{out}}^{\max}(s_{\text{out},j'}) y\left(s_{\text{in},j'}, p-2\right) \\ &\quad + \sum_{j'} mstu_{\text{out}}\left(j',j,p-1\right) cst_{\text{out}}\left(j',p-1\right) + M\left(s_{\text{in}}\right) y\left(s_{\text{in},j}, p-1\right) \\ &\forall j,j' \in J,\ s_{\text{in},j}, s_{\text{in},j'} \in S_{\text{in},j},\ s_{\text{out},j}, s_{\text{out},j'} \in s_{\text{out},j},\ p \in P,\ p > p_2 \end{aligned} \tag{9.77}$$

There are two forms of nonlinearities in constraint (9.77). The first comprises of a continuous variable and a binary variable and the second comprises of two continuous variables. The first nonlinearity can be linearised exactly using a Glover transformation (1975) and the second can be linearised using a relaxation-linearization technique proposed by Quesada and Grossman (1995), where necessary.

The given constraints above complete the mathematical formulation. Following are the solution procedures followed to solve the two problems considered.

9.4 Solution Procedures for the Problems Considered in the Inherent Storage Methodology

The solution procedure for the first problem is different to that of the second problem. The solution procedure for the first problem is a two step procedure. In the first step of the solution procedure the minimum wastewater target is determined considering infinite central storage and no inherent storage. All the constraints accounting for inherent storage are removed from the model in this step. In the second step the amount of freshwater used is set to the minimum determined in the first step and inherent storage is included. The objective in this step is to determine the minimum size of the central storage vessel while still achieving the minimum freshwater target set in step one. The objective function used in the second step is given in constraint (9.78).

$$\min \sum_{p \in P} q(p) \tag{9.78}$$

The solution procedure for the second problem is a one step procedure in which the problem is solved considering both central storage and inherent storage. The central storage vessel in this case has a fixed size. The objective used is dependent on whether the required production is known or not. If the required production is known then the objective is the minimisation of effluent. If this is not known then the objective is the maximisation of profit.

The use of these procedures is demonstrated in the illustrative examples considered below.

9.5 Illustrative Examples Using Inherent Storage

Two illustrative examples are presented to demonstrate the application of the inherent storage methodology. In the first example the objective is to minimise the amount of wastewater generated as well as the size of central storage required. In the second example the objective is to minimise wastewater while making use of inherent storage and central storage.

9.5.1 First Illustrative Example

The case study presented by Majozi (2006) is used as the basis for the first illustrative example. The plant considered involves 5 operations producing three products. Wastewater containing a single contaminant, namely NaCl, is produced from each of the 5 operations. In the first operation, operation A, a reaction takes place in an organic solvent that is immiscible with water. In this operation the NaCl formed is removed from the organic phase through liquid–liquid extraction with water as the extracting solvent. Water is used in operations B and D as a reaction solvent. The intermediate products produced from operations B and D are sent to operations C and E, both polishing operations in which water is used to extract any remaining NaCl. Operations C and E serve as quality control operations. Operation A produces product 1 and operations C and E produce products 2 and 3 respectively.

Due to the fact that operations C and E serve as quality control operations, the mass load removed by each is essentially zero. However, each operation requires a minimum of 300 kg water and a maximum of 400 kg of water. The concentration data, water requirements and starting and ending times of each operation are given in Table 9.1. The capacity of each of the units used for the required operations is 2000 kg, with each operation taking place in a dedicated processing unit.

As mentioned previously the objective was the minimisation of the required storage vessel while still achieving the minimum wastewater target. This was done using the solution algorithm discussed earlier.

The model used in the first step was based on the formulation given above, however, all the constraints and terms concerned with the inherent storage were removed. The model for the first step thus takes on the familiar form of the wastewater minimisation formulation for single contaminants. The central storage in this step has no limit on its capacity. The reformulated constraint (9.77) was used in this step without the term for the inherent storage. The only term that was linearised in this model was a bilinear term involving the binary variable and continuous variable in constraint (9.77). This was done using a Glover transformation (1975). The remaining bilinear terms in the model could not be linearised due to the unlimited size of the central storage vessel. The model for the second step comprised of all the constraints given above including all the terms and constraints dealing with inherent storage. Due to the size of the storage being variable the model could not be fully linearised.

Table 9.1 Data for first illustrative example using inherent storage

Operation (j)	Mass load (kg)	$CI_{\mathrm{in}}^{\max}(j)$ (kg salt/kg water)	$CI_{\mathrm{out}}^{\max}(j)$ (kg salt/kg water)	Mass water (kg)	Start time (h) $\left(t_u\left(s_{\mathrm{in},j},p\right)\right)$	End time (h) $\left(t_p\left(s_{\mathrm{out},j},p\right)\right)$
A	100	0	0.1	1000	0	3
B	72.8	0.25	0.51	280	0	4
C	0	0.1	0.1	[300,400]	4	5.5
D	72.8	0.25	0.51	280	2	6
E	0	0.1	0.1	[300,400]	6	7.5

The resulting model for both steps was solved in GAMS 22.0 using the DICOPT2 solution algorithm. The MIP solved was CPLEX 9.1.2 and the NLP solver used was CONOPT3. For the first step the resulting model had 280 binary variables and had a solution time of 47.43 CPU seconds using an Intel Duo Core 1.66 GHz processor. DICOPT required three major iterations to find a solution in which the minimum wastewater target was 1285.5 kg, a 45% reduction in wastewater. The required storage capacity to achieve this was 800 kg.

The resulting model for the second step had 630 binary variables with a solution time of 4.78 CPU seconds using the same processor as previously. The wastewater target identified in the first step was used as a fixed amount in this step. In finding the solution DICOPT required three major iterations. The final size of the central storage vessel in this step was zero. This means that the minimum wastewater target can be achieved using inherent storage exclusively. In both solutions the optimal number of time points was 7. This was found using an iterative procedure in which the number of time points is increased by one and the problem solved until the objective function attained a constant value for two or more iteration steps.

Using the methodology derived by Majozi (2006) a central storage vessel with a minimum capacity of 300 kg was identified. However, as can be seen above, with the use of inherent storage this is not required. In both the solution found by Majozi (2006) and that above the minimum wastewater target identified was 1285.5 kg.

The resulting Gantt chart is shown is Fig. 9.2. Figure 9.2 depicts the final schedule of the operations including inherent storage. The bold numbers in the figure

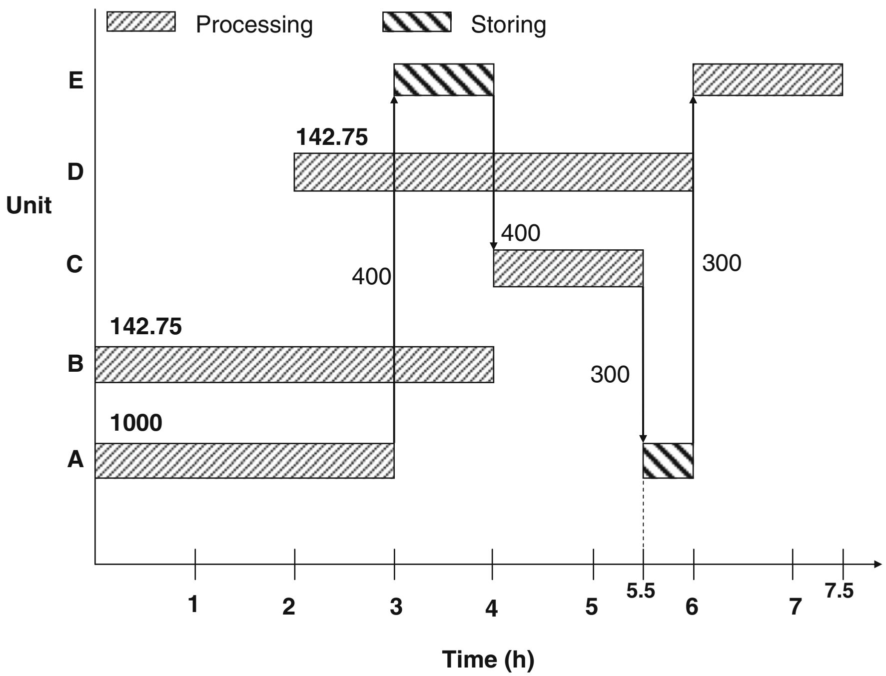


Fig. 9.2 Resulting Gantt chart for first illustrative example using inherent storage (Gouws and Majozi, 2009)

represent freshwater used and the normal numbers represent wastewater reuse. As can be seen the processing units in which operations A and E take place each serve as a storage vessel. A portion of the wastewater generated by operation A is sent to the unit in which operation E takes place. This water is later used by operation C. A portion of the wastewater from operation C is sent for storage to the vessel in which operation A takes place. This water is later used by operation E.

9.5.2 Second Illustrative Example

In the second example a plant comprising of four water using operations is considered, each producing a different product and each occurring in a separate unit. It is required that operation 1 operates twice and operations 2, 3, and 4 operate once within the 12 h time horizon. It is required that operation 2 starts at the beginning of the time horizon. A further requirement is that operation 3 always occurs before operation 4. However, the starting time of operation 4 need not coincide with the ending time of operation 3.

The maximum inlet and outlet concentrations, maximum allowable water, mass load and duration of each operation are given in Table 9.2. The central storage vessel considered has a capacity of 500 kg and each unit, in which each operation occurs, has a capacity of 2000 kg.

The second example was solved for two cases. In the first case wastewater storage was only available in the form of a central storage vessel, no inherent storage was included. In the second case wastewater storage was available in the form of inherent storage and a central storage vessel. In both instances the objective function was the minimisation of effluent, since the production was known.

Solution for Second Illustrative Example with Central Storage Only

The constraints that comprised the model in this instance were the constraints given above excluding the constraints and terms dealing with inherent storage. The resulting model contained a number of nonlinearities and was thus solved using the solution algorithm proposed by Gouws et al. (2008). The proposed solution algorithm is a two step procedure in which the exact nonlinear model is linearised using the relaxation-linearization technique proposed by Quesada and Grossman (1995)

Table 9.2 Data for second illustrative example using inherent storage

Unit	Max inlet concentration (g/kg water)	Max. outlet concentration (g/kg water)	Mass load (g)	Max. allowable water (kg)	Duration (h)
1	0.005	0.065	75	1250	4.5
2	0	0.010	10	1000	2
3	0.020	0.050	15	500	3.75
4	0.01	0.015	10	2000	5.5

resulting in a linear model. The linear model is solved to attain an initial solution for the exact nonlinear model, which is subsequently solved.

The resulting MILP and MINLP used in the solution algorithm were both formulated in GAMS 22.0. In the case of the MILP, CPLEX 9.1.2 was used as the solver and in the case of the MINLP, the DICOPT2 solution algorithm was used. In the DICOPT solution algorithm CPLEX 9.1.2 was used as the MIP solver and CONOPT3 as the NLP solver.

The number of binary variables for both the MILP and MINLP was 112 with 4 time points. The solution was found in 0.89 CPU seconds using the same processor as in the previous example. The objective function had a final value of 3684.61 kg of water. If wastewater recycle/reuse had not been considered the amount of effluent would have been 4274.40 kg, thus a wastewater reduction of 13.8% is achieved. The value of objective function for MILP and MINLP were not the same in this case, which means the final solution obtained is not globally optimal (Gouws et al., 2008). The value of the objective function from the MILP was only marginally lower being 3678.84 kg.

The resulting schedule is shown in Fig. 9.3. The bold numbers in Fig. 9.3 represent freshwater used and the normal numbers wastewater reused. One would notice in the figure that operation 2 sends 500 kg of wastewater to the central storage vessel. Operation 1 uses all of the 500 kg of water. Operation 4 directly reuses 500 kg of wastewater from operation 2.

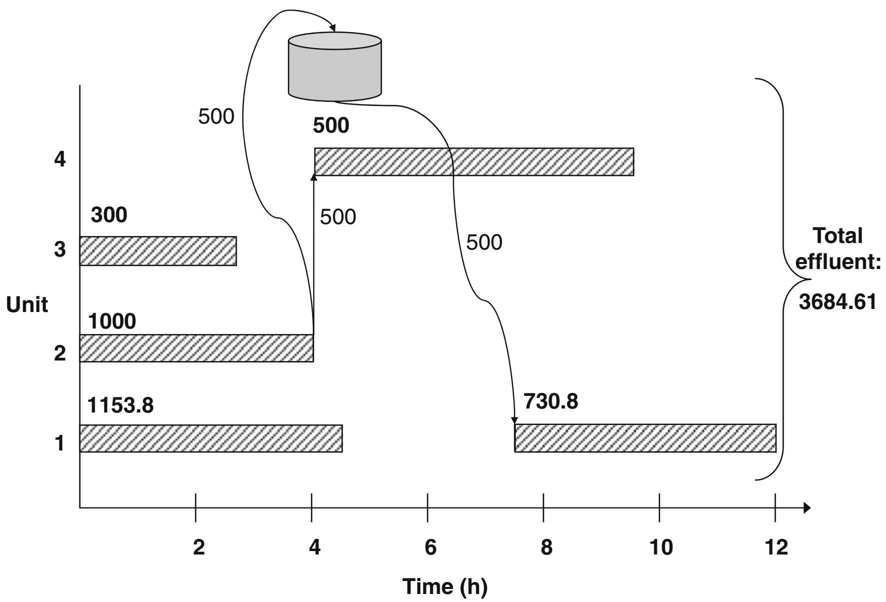


Fig. 9.3 Resulting schedule for second illustrative example with central storage only (Gouws and Majozi, 2009)

Solution to Second Illustrative Example with Inherent Storage and Central Storage

As with the previous instance the example was solved using the solution algorithm proposed by Gouws et al. (2008). The resulting MILP and MINLP solved using GAMS 22.0. In the case of the MILP the solver used was CPLEX 9.1.2 and in the case of the MINLP the DICOPT2 solution algorithm was used. The MIP solver used in the DICOPT algorithm was CPLEX 9.1.2 and the NLP solver was CONOPT3.

The resulting number of binary variables for both the MILP and MINLP was 360 with 6 time points. The optimal value of the objective function for both the MILP and MINLP was 3620.52 kg of water, which means that the solution obtained is globally optimal (Gouws et al., 2008). This is a lower objective function value than that achieved in the previous case. In this case an improved savings of 15.3% is achieved. The final solution was attained in 896.37 CPU seconds using the same processor as in the first illustrative example.

The resulting Gantt chart is shown in Fig. 9.4. The bold numbers in the figure represent freshwater used and the normal numbers represent wastewater reused. As one would notice from the figure, the unit in which operation 2 takes place serves as a wastewater storage vessel. In this unit 1000 kg of wastewater is stored, which is later used by operation 1 and operation 4. 675 kg of the stored water is used by operation 1 and 375 kg is used by operation 4. Important to note from Fig. 9.4 is

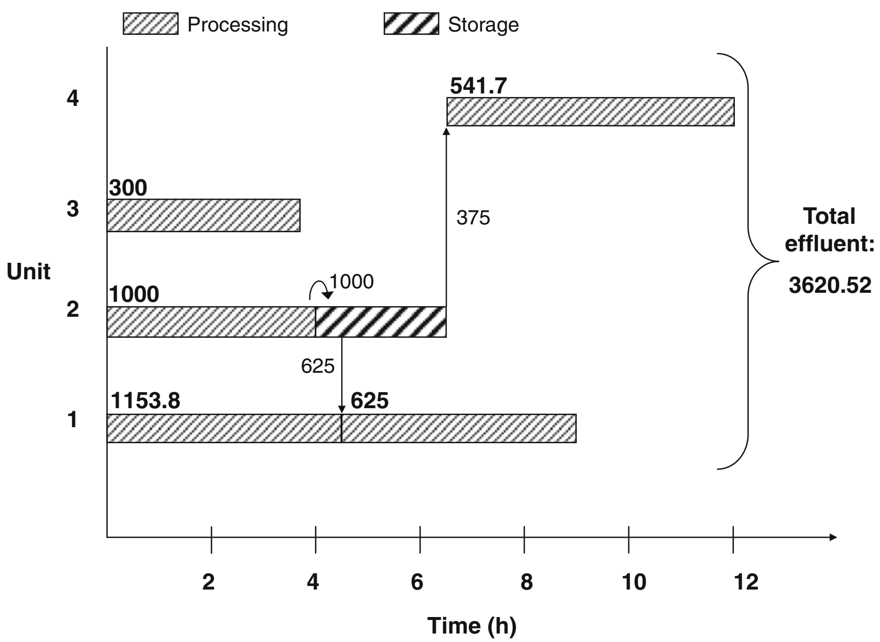


Fig. 9.4 The resulting schedule for second illustrative example with inherent storage and central storage (Gouws and Majozi, 2009)

that the central storage vessel is not used at all, whilst an improved solution to the previous solution has been found.

The solutions obtained from the second illustrative example show the advantages of inherent storage. In the first case a higher minimum wastewater target was found, namely 3684.61 kg, when compared to the second case. This is because the size of the central storage vessel limited the reuse possibilities. In the second case the usage of inherent storage, in addition to central storage, allowed for further wastewater storage opportunities and thus a lower objective function value was found, namely 3620.52 kg. The difference is observed due to the fact that different reuse opportunities are exploited in the second solution, which allow for the production of less effluent. The better solution in the second case is further substantiated since the solution obtained from the relaxed MILP has a higher objective function value than that in the second solution. In first solution, the objective function value for the MILP provides a lower bound to the exact MINLP.

A note must be made on the solution time of the second case in the second illustrative example. The solution time is excessive, approximately 900 CPU seconds. The long solution time is due to the solution procedure used. The solution time for the MILP accounted for more than 99% of the total solution time. Shorter solution times might have been achieved if a different solution procedure had been followed, e.g. only partially linearising the MINLP. However, the final solution found is globally optimal, which justifies the usage of the solution procedure.

In both solutions presented above, cyclic operation has not been investigated. However, it is very possible that if the operation were cyclic further reuse opportunities could present themselves.

9.6 Conclusions

The methodology presented above makes use of idle processing units to provide additional storage opportunities for wastewater. The use of the idle processing units also allows for the minimisation of central storage for wastewater.

The methodology applied to two types of problems. In the first problem the size of the central storage vessel was minimised and at the same time the effluent produced was also minimised. This was done in a two step procedure where the problem is first solved without inherent storage and unlimited intermediate storage. The objective of this step is to determine the minimum wastewater target. The second step involves the minimisation of the size of the central storage vessel. The wastewater target set in the previous step is fixed in this step. The objective function in this case is the minimisation of storage. The second problem is the general wastewater minimisation problem where inherent storage is included, either with a central storage vessel or without.

The methodology was applied to two illustrative examples. The first example stems from a case study presented by Majozi (2006). This example involved 5 processes, with a fixed schedule. The objective in the example was to minimise the size of the central storage vessel and minimise the amount of wastewater generated. The minimum wastewater target identified was 1285.5 kg and no storage central vessel was required to achieve this since the idle processing units provided sufficient wastewater storage capacity. The wastewater storage capacity in the form of inherent storage was 700 kg.

In the second illustrative example a small operation with four processing units was considered. The objective of the example was to minimise the amount of wastewater produced. The problem was solved first without inherent storage and a central storage vessel and then with both central storage and inherent storage. In the first scenario a wastewater target of 3684.61 kg was identified using central storage. In the second scenario a lower wastewater target was identified using inherent storage and central storage. In the second case a wastewater target of 3620.52 kg was identified. Furthermore, the solution obtained from the second case is globally optimal since the objective function of the relaxed MILP equalled that of the exact MINLP.

The main drawback of the methodology is that cross transfer of material is still possible. Cross transfer occurs when a unit is sending and receiving material at the same time. If one considers the physical meaning of this, one would then be mixing the incoming material with the material that is leaving the vessel. This is a general problem with MINLP and MILP scheduling formulations and research is still underway to find a solution.

9.7 Exercise

Task: Apply the mathematical formulation presented in this chapter to verify the results presented for the first and second illustrative examples. In what types of problems is the proposed approach less likely to yield good results?

References

Glover, F., 1975. Improved linear integer programming formulation of nonlinear problems. Manag. Sci., 22(4): 455–460

Gouws, J.F., Majozi, T., Gadalla, M., 2008. Flexible mass transfer model for water minimization in batch plants. Chem. Eng. Process., 47: 2323–2335

Gouws, J.F., Majozi, T., 2009. Usage of inherent storage for minimisation of wastewater in multipurpose batch plants. Chem. Eng. Sci., 64: 3545–3554

Majozi, T., 2005. Wastewater minimization using central reusable storage in batch plants. Comput. Chem. Eng., 29: 1631–1646

Majozi, T., 2006. Storage design for maximum wastewater reuse in multipurpose batch plants. Ind. Eng. Chem. Res., 45: 5936–5943

Majozi, T., Zhu, X., 2001. A novel continuous-time MILP formulation for multipurpose batch plants.1. Short-term scheduling. Ind. Eng. Chem. Res., 40(23): 5935–5949

Quesada, I., Grossmann, I.E., 1995. Global optimization of bilinear process networks with multicomponent flows. Comput. Chem. Eng., 19: 1219–1242

Savelski, M., Bagajewicz, M., 2000. On the optimality conditions of water utilization systems in process plants with single contaminants. Chem. Eng. Sci. 55: 5035–5048

Chapter 10
Heat Integration in Multipurpose Batch Plants: I. Direct Heat Integration

Overview The foregoing 9 chapters of this book have only dealt with the mass aspects of process integration without any consideration for energy aspects. As batch processes become more popular, it has become necessary to develop energy optimisation techniques that are particular to batch plants. This chapter presents a mathematical formulation that is founded on uneven discretization of the time horizon for direct heat integration of multiproduct and multipurpose batch plants. Direct heat integration implies that the operations that are involved in exchanging energy occur at almost the same time or within a common time interval. It is also assumed that the thermal driving forces and the duties are compatible. The presented technique allows for external energy to be supplied supplementing any deficiency during heat integration. Two cases are considered. In the first case the energy requirement of each process is dependent on the amount of material that is allowed to vary throughout the time horizon of interest. In the second case, the amount of material processed in each unit is fixed whenever the task occurs during the time horizon, thereby fixing the energy requirement. Unlike the second case which is inherently linear, the mathematical formulation that corresponds to the first case is nonlinear, but is linearized exactly to yield a globally optimal solution. An illustrative example and a case study are presented to facilitate understanding. In addition, an exercise that is based on the presented case study is provided for the convenience of the reader.

10.1 Some Background to Heat Integration of Batch Plants

As highlighted in Chapter 1 of this textbook, process integration in general has always been the privilege of continuous rather than batch chemical processes. However, the last 2 decades have been characterised by heightened interest in understanding batch processes and developing techniques that are particular to these operations. Some of the very early work in batch energy process integration was proposed by Vaselanak et al. (1986) where heat integration of batch vessels containing hot and cold fluids was addressed. Four cases were considered. In the first case, the fluid from one vessel was allowed to return to the same vessel after exchanging heat with the fluid of another vessel via a common heat exchanger. In the second

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_10,

case, a heating or cooling medium was used to transfer heat between the hot fluid vessel and the cold fluid vessel, thereby maintaining the heat integrated fluids within the vessels throughout the heat exchange process. The third case entailed the transfer of fluids from their original vessels to receiving vessels while being heated or cooled. The fourth case was the combination of the above cases. Implicit in their analysis was the given schedule of the operations. A heuristic procedure was proposed for the cases where the final temperatures were not limiting and an MILP formulation for the cases where the final temperatures were limiting. Subsequent to this, other mathematical formulations for heat integration of batch processes have been proposed by Peneva et al. (1992), Ivanov et al. (1992), Corominas et al. (1993), Papageorgiou et al. (1994), Vaklieva-Bancheva et al. (1996) and Adonyi et al. (2003).

Peneva et al. (1992) and Ivanov et al. (1992) addressed the problem of designing a minimum total cost heat exchanger network for given pair wise matches of batch vessels. An implicit predefined schedule was also assumed. Corominas et al. (1993) considered the problem of designing a minimum cost heat exchanger network and a heat exchange strategy for multiproduct batch plants operating in a campaign mode. The objective was to maximize heat exchange in a pre-specified campaign of product batches with hot streams requiring cooling and cold streams requiring heating. The emphasis on campaign mode implies that the proposed methodology cannot be applied in situations where equipment scheduling is of essence. Papageourgiou et al. (1994) extended the discrete-time formulation of Kondili et al. (1993) for scheduling of multipurpose batch plants by including heat integration aspects. Direct and indirect heat integration configurations were addressed. The main drawback of all discrete-time formulations is their explosive binary dimension, which requires enormous computational effort. Vaklieva-Bancheva et al. (1996) improved the work of Ivanov et al. (1992) by embedding the heat integration framework within an overall scheduling framework. However, the authors only addressed a special case in which the plant is assumed to operate in a zero-wait overlapping mode, where each product must pass through a subset of the equipment stages, and production is organized in a series of long campaigns. Recently, Pinto et al. (2003) presented a discrete-time mixed integer mathematical formulation for the design of heat integrated multipurpose plants based on superstructure approach. A graph theory based technique that incorporates heat integration within scheduling of multipurpose plants has also been proposed by Adonyi et al. (2003) with emphasis on make span minimization.

Other established attempts on heat integration of batch plants are based on the concept of pinch analysis (Linnhoff et al., 1979; Umeda et al., 1979), which was initially developed for continuous processes at steady-state. As such, these methods assume a pseudo-continuous behaviour in batch operations either by averaging time over a fixed time horizon of interest (Linnhoff et al., 1988) or assuming fixed production schedule within which opportunities for heat integration are explored (Kemp and MacDonald, 1987, 1988; Obeng and Ashton, 1988; Kemp and Deakin, 1989). These methods cannot be applied in situations where the optimum schedule has to be determined simultaneously with the heat exchanger network that minimises external energy use.

In this chapter a technique that is applicable to batch plants, taking into account the time dependent nature of tasks involved in a batch plant, is presented. This work has also been presented in a journal publication (Majozi, 2006).

10.2 Problem Statement

The problem addressed in this chapter can be stated as follows. Given:

(i) production scheduling data, i.e. equipment capacities, task durations, time horizon of interest, recipe for each product as well as cost of raw materials and selling price of final products,
(ii) hot and cold duties for tasks that require heating and cooling, respectively and
(iii) cost of cooling water and steam,

determine the production schedule that is concomitant with maximum process–process heat transfer and maximum profit. In the context of this chapter, profit is defined as the difference between revenue and operating costs. The latter constitute raw material costs and external utility (cooling water and steam) costs. It is assumed that sufficient temperature driving forces exist between matched tasks for process–process heat transfer. Also, each task is allowed to operate either in an integrated or standalone mode. If heat integration cannot supply sufficient duty, external utility is supplied to complement the deficit. Whilst heat integrated tasks have to be active within a common time interval to effect direct heat transfer, they need not necessarily commence nor end at the same time. Moreover, the heat integrated tasks can either belong to the same process or distinct processes within reasonable proximity.

10.3 Mathematical Formulation

The presented mathematical formulation is made up of the following sets, variables and parameters.

Sets

J	$\{j \mid j \text{ is a unit}\}$
J_{h}	$\{j_h \mid j_h \text{ is a unit that requires heating}\} \subseteq J$
J_{c}	$\{j_c \mid j_c \text{ is a unit that requires cooling}\} \subseteq \mathrm{J}$
p	$\{p \mid p \text{ is a time point}\}$
$S_{\mathrm{in},j}$	$\{S_{\mathrm{in},j} \mid S_{\mathrm{in},j} \text{ is an input state to unit j}\}$
$S_{\mathrm{out},j}$	$\{S_{\mathrm{out},j} \mid S_{\mathrm{out},j} \text{ is an output state from unit j}\}$

Variables

$m_u\,(s,p)$	amount of state s used at time point p, $s \in S_{\text{in},j}$
$q\,(j,p)$	amount of external utility required by unit j when operating in a standalone mode at time point p
$q'\,(j,p)$	amount of external utility required by unit j when operating in a heat-integrated mode at time point p
$q_h\left(j,j',p\right)$	amount of heat exchanged by heat-integrated units j and j' at time point p
$t_p\,(s,p)$	time at which state s is produced at time point p, $s \in S_{\text{out},j}$
$t_u\,(s,p)$	time at which state s is used at time point p, $s \in S_{\text{in},j}$
$x\left(j,j',p\right)$	binary variable associated with heat integration of units j and j' at time point p
$y\,(s,p)$	binary variable associated with usage of state s at time point $p, s \in S_{\text{in},j}$

Glover Transformation Variables

$$\Gamma_1\left(s_{\text{in},j},p\right);\;\; \Gamma_2\left(j,j',p\right);\;\; \Gamma_3\left(s_{\text{in},j},p\right);\;\; \Gamma_4\left(j,j',p\right)$$

Parameters

H	time horizon of interest
$K_{1,j}; K_{2,j}$	coefficients for external utility requirements in unit j when operated in a standalone mode
$K'_{1,j}; K'_{2,j}$	coefficients for external utility requirements in unit j when operated in a heat-integrated mode
$Q_{cu}\,(j)$	fixed amount of cold utility requirement in unit j
$Q_{hu}\,(j)$	fixed amount of hot utility requirement in unit j
$Q^U_{hi}\left(j,j'\right)$	maximum amount of heat exchanged between units j and j'
$Q^L_{hi}\left(j,j'\right)$	minimum amount of heat exchanged between units j and j'
$\tau\left(s_{\text{in},j}\right)$	fixed duration in unit j when operated in a standalone mode
$\tau'\left(s_{\text{in},j}\right)$	fixed duration in j when operated in a heat integrated mode
V^U_j	maximum capacity for unit j

The presented mathematical formulation is an extension of the scheduling model proposed by Majozi and Zhu (2001), which uses a state sequence network (SSN) representation. This formulation is based on an uneven discretization of time framework (see Chapter 2, Fig. 2.4) as shown in Fig. 10.1. A time point corresponds to the beginning of a particular task and is not necessarily equidistant from the preceding and the succeeding time points, as it encountered in discrete-time formulations. The

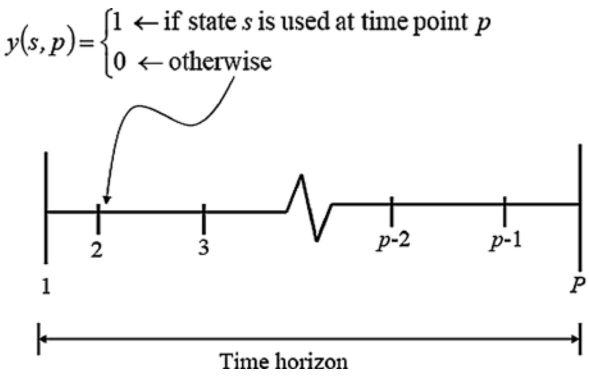


Fig. 10.1 Uneven discretization of time horizon

details of the scheduling formulation without heat integration will not be presented in this chapter as they appear in Chapter 2. Two cases are considered. The first case involves a situation where the batch size is allowed to vary at different instances along the time horizon of interest. The second case is based on fixed batch sizes.

10.3.1 Case 1: Variable Batch Size

$$\sum_{j' \in J_c} x\left(j,j',p\right) \le y\left(s_{\text{in},j},p\right), \quad \forall p \in P,\ j \in J_h, s_{\text{in},j} \in S_{\text{in},j} \tag{10.1}$$

$$\sum_{j' \in J_h} x\left(j,j',p\right) \le y\left(s_{\text{in},j'},p\right), \quad \forall p \in P,\ j \in J_c, s_{\text{in},j'} \in S_{\text{in},j} \tag{10.2}$$

Constraints (10.1) states that if unit j, which requires heating, is integrated with any unit j' requiring cooling at time point p, then unit j must be active at that time point. However, the fact that unit j is active at time point p does not necessarily mean that it operates in an integrated mode, as standalone operation is allowed. Constraints (10.2) is similar to constraints (10.1), but applies to unit j'. It is also worthy of note that the above constraints ensure a one-to-one heat integration arrangement between heat integrated units. This is indeed a preferred option for plant operability purposes.

$$\begin{aligned} q\left(j,p\right) = {} & K_{1,j}\left(y\left(s_{\text{in},j},p\right) - \sum_{j' \in J_c} x\left(j,j',p\right)\right) \\ & + K_{2,j}\sum_{s_{\text{in},j}} m_u\left(s_{\text{in},j},p\right)\left(y\left(s_{\text{in},j},p\right) - \sum_{j' \in J_c} x\left(j,j',p\right)\right), \\ & \forall p \in P, j \in J_h, s_{\text{in},j} \in S_{\text{in,j}} \end{aligned} \tag{10.3}$$

$$q'(j,p) = K'_{1,j} \sum_{j' \in J_c} x(j,j',p) + K'_{2,j} \sum_{s_{\text{in},j}} m_u(s_{\text{in},j},p) \sum_{j' \in J_c} x(j,j',p), \quad \forall p \in P, j \in J_h, s_{\text{in},j} \in S_{\text{in},j} \tag{10.4}$$

Constraints (10.3) and (10.4) are expressions of the external utility requirement for unit j, which requires heating, when operating in a standalone and integrated mode, respectively.

$$q(j',p) = K_{1,j'} \left(y(s_{\text{in},\,j'},p) - \sum_{j' \in J_h} x(j,j',p) \right) + K_{2,j'} \sum_{s_{\text{in},j}} m_u(s_{\text{in},\,j'},p) \left(y(s_{\text{in},\,j'},p) - \sum_{j' \in J_h} x(j,j',p) \right), \quad \forall p \in P, j' \in J_h, s_{\text{in},j'} \in S_{\text{in},j} \tag{10.5}$$

$$q'(j',p) = K'_{1,j'} \sum_{j \in J_h} x(j,j',p) + K'_{2,j'} \sum_{s_{\text{in},j}} m_u(s_{\text{in},j'},p) \sum_{j \in J_{hc}} x(j,j',p), \quad \forall p \in P, j' \in J_h, s_{\text{in},j'} \in S_{\text{in},j} \tag{10.6}$$

Constraints (10.5) and (10.6) are similar to constraints (10.3) and (10.4), but apply to unit j', which requires cooling, when operating in a standalone and integrated mode, respectively. It is evident that constraints (10.3)–(10.6) involve bilinear terms comprising a continuous and a binary variable. These terms can be exactly linearized using Glover transformation to yield an overall model for which global optimality can be guaranteed as shown below.

Linearization Using Glover Transformation

The bilinear terms in constraints (10.3) and (10.4) are replaced by variables Γ_1 and Γ_2 as shown in constraints (10.7) and (10.8). These variables are the linearly defined in constraints (10.11), (10.12), (10.12) and (10.14). Variables Γ_3 and Γ_4 replacing the bilinear terms in constraints (10.5) and (10.6) are introduced in constraints (10.9) and (10.10). The linear definitions of these variables are similar to those for Γ_1 and Γ_2.

$$q(j,p) = K_{1,j} \left(y(s_{\text{in},j},p) - \sum_{j' \in J_c} x(j,j',p) \right) + K_{2,j} \left(\Gamma_1(s_{\text{in},j},p) - \Gamma_2(j,j',p) \right), \quad \forall p \in P,\ j \in J_h, s_{\text{in},\,j} \in S_{\text{in},\,j} \tag{10.7}$$

$$q'(j,p) = K'_{1,j} \left(\sum_{j' \in J_c} x(j,j',p) \right) + K'_{2,j} \Gamma_2(j,j',p), \quad \forall p \in P, j \in J_h, s_{\text{in},j} \in S_{\text{in},j} \tag{10.8}$$

$$q\left(j',p\right) = K_{1,j'}\left(y\left(s_{\text{in},j'},p\right) - \sum_{j'\in J_h} x\left(j,j',p\right)\right) + K_{2,j'}\left(\Gamma_3\left(s_{\text{in},j'},p\right) - \Gamma_4\left(j,j',p\right)\right), \quad \forall p \in P, j' \in J_c, s_{\text{in},j'} \in S_{\text{in},j} \tag{10.9}$$

$$q'\left(j',p\right) = K'_{1,j'}\left(\sum_{j'\in J_h} x\left(j,j',p\right)\right) + K'_{2,j'}\Gamma_4\left(j,j',p\right), \quad \forall p \in P, j' \in J_c, s_{\text{in},j'} \in S_{\text{in},j} \tag{10.10}$$

$$\sum_{s_{\text{in},j}} m_u\left(s_{\text{in},j},p\right) - V_j^U\left(1 - y\left(s_{\text{in},j},p\right)\right) \le \Gamma_1\left(s_{\text{in},j},p\right) \le \sum m_u\left(s_{\text{in},j},p\right), \quad \forall p \in P, j \in J_h, s_{\text{in},j} \in S_{\text{in},j} \tag{10.11}$$

$$0 \le \Gamma_1\left(s_{\text{in},j},p\right) \le V_j^U y\left(s_{\text{in},j},p\right), \quad \forall p \in P, j \in J_h, s_{\text{in},j} \in S_{\text{in},j} \tag{10.12}$$

$$\sum_{s_{\text{in},j}} m_u\left(s_{\text{in},j},p\right) - V_j^U\left(1 - \sum_{j'\in J_c} x\left(j,j',p\right)\right) \le \Gamma_2\left(s_{\text{in},j},p\right) \le \sum_{s_{\text{in},j}} m_u\left(s_{\text{in},j},p\right), \quad \forall p \in P, j \in J_h, s_{\text{in},j} \in S_{\text{in},j} \tag{10.13}$$

$$0 \le \Gamma_2\left(s_{\text{in},j},p\right) \le V_j^U \sum_{j'\in J_c} x\left(j,j',p\right), \quad \forall p \in P, j \in J_h, s_{\text{in},j} \in S_{\text{in},j} \tag{10.14}$$

Constraints (10.1), (10.2), (10.8)–(10.14), in conjunction with the overall plant scheduling constraints, constitute a complete MILP formulation for direct heat integration in batch processes in a situation where the batch size is allowed to vary at different instances along the time horizon of interest.

10.3.2 Case 2: Fixed Batch Size

In a situation where the batch size processed in a particular unit at various time points along the time horizon is fixed, the heat duty requirement will also be fixed. Therefore, it is specified as a parameter rather than a variable. Constraints (10.1) and (10.2) still hold in this scenario. However, the following additional constraints are also necessary:

$$Q_{cu}\left(j'\right) y\left(s_{\text{in},j'},p\right) = q\left(j',p\right) + \sum_{j\in J_h} q_{\text{hi}}\left(j,j',p\right), \quad \forall p \in P, j' \in J_c, s_{\text{in},j} \in S_{\text{in},j} \tag{10.15}$$

$$Q_{\text{hu}}(j)\, y\left(s_{\text{in},j}, p\right) = q\,(j,p) + \sum_{j' \in J_c} q_{\text{hi}}\left(j,j',p\right), \qquad \forall p \in P, j \in J_h, s_{\text{in},j} \in S_{\text{in},j} \tag{10.16}$$

$$Q_{\text{hi}}^{L}\left(j,j'\right) x\left(j,j',p\right) \le q_{\text{hi}}\left(j,j',p\right) \le Q_{\text{hi}}^{U}\left(j,j'\right) x\left(j,j',p\right), \quad \forall p \in P, j \in J_h, j' \in J_c \tag{10.17}$$

Constraints (10.15) states that the amount of cold duty required by unit j' at any point along the time horizon of interest is comprised of external cold utility and cold duty from heat integration with another unit j. Constraints (10.16) is similar to constraints (10.15) and applies to unit j requiring heating. Constraints (10.17) is a feasibility constraints, which ensures that in the absence of heat integration all the heat duty requirements of either unit j or j' are satisfied by external utilities. The upper bound on the amount of heat exchanged between unit j and unit j' will always be the minimum of the required cold and hot utilities as captured in Constraint (10.17′).

$$Q_{\text{hi}}^{U}\left(j,j'\right) = \min_{\substack{j \in J_h \\ j' \in J_c}} \left\{Q_{\text{cu}}\left(j'\right), Q_{\text{hu}}(j)\right\} \tag{10.17′}$$

The lower bound, on the other hand, is at the discretion of the designer. Constraints (10.1), (10.2) and (10.15)–(10.17′), in conjunction with the overall plant scheduling constraints, constitute a complete MILP formulation for direct heat integration in batch processes in a situation where the batch size is fixed along the time horizon of interest.

In order to ensure that the heat-integrated units are active within a common time interval, the following con constraints is necessary. In constraints (10.18), unit j has a relatively longer duration time than unit j'. If duration times are equal, then constraints (10.19) and (10.20) are necessary.

$$t_p\left(s_{\text{out},j}, p\right) \ge t_p\left(s_{\text{out},j'}, p\right) - H\left(1 - x\left(j,j',p\right)\right), \qquad \forall p \in P, j \in J_h, j' \in J_c \tag{10.18}$$

$$t_p\left(s_{\text{out},j}, p\right) \ge t_p\left(s_{\text{out},j'}, p\right) - H\left(1 - x\left(j,j',p\right)\right), \qquad \forall p \in P, j \in J_h, j' \in J_c \tag{10.19}$$

$$t_p\left(s_{\text{out},j}, p\right) \le t_p\left(s_{\text{out},j'}, p\right) + H\left(1 - x\left(j,j',p\right)\right), \qquad \forall p \in P, j \in J_h, j' \in J_c \tag{10.20}$$

In a situation where duration time varies with the mode of operation, the duration constraints has to be modified as shown in constraints (10.21) and (10.22) for units j and j', respectively.

$$t_p\left(s_{\text{out},j},p\right)=t_u\left(s_{\text{in},j},p\right)+\tau\left(s_{\text{in},j}\right)\left(y\left(s_{\text{in},j},p\right)-\sum_{j'\in J_c}x\left(j,j',p\right)\right)$$
$$+\ \tau'\left(s_{\text{in},j}\right)\sum_{j'\in J_c}x\left(j,j',p\right),\quad \forall p\in P, j\in J_h, s_{\text{in},j}\in S_{\text{in},j}, s_{\text{out},j}\in S_{\text{out},j} \tag{10.21}$$

$$t_p\left(s_{\text{out},j'},p\right)=t_u\left(s_{\text{in},j'},p\right)+\tau\left(s_{\text{in},j'}\right)\left(y\left(s_{\text{in},j'},p\right)-\sum_{j'\in J_h}x\left(j,j',p\right)\right)$$
$$+\,\tau'\left(s_{\text{in},j'}\right)\sum_{j\in J_h}x\left(j,j',p\right),\ \forall p\in P, j'\in J_c, s_{\text{in},j'}\in S_{\text{in},j}, s_{\text{out},j'}\in S_{out,j} \tag{10.22}$$

The performance of the presented formulation was tested by applying it to a literature example and an industrial case study. All solutions were obtained using the GAMS/CPLEX solver in a 1.4 GHz Pentium M processor.

10.4 Literature Example

The following example has been extracted directly from literature (see Chapter 2) and is presented to demonstrate the effectiveness of the proposed formulation. It entails a plant manufacturing two products, *Product*1 and *Product*2, according to the following recipe which corresponds to the STN and SSN shown in Fig. 10.2. The recipe as well as heat requirements are shown in Table 10.1. The minimum allowable size of batches processed in the reactor and the column cannot be less than 25% of their normal capacity. The corresponding figure for the filter is 10%. Sufficient dedicated storage is provided for all raw materials and final products, while storage vessels of 100 t are available for each of the two intermediates. Unlimited availability of steam and cooling water is assumed, with corresponding unit costs of 200 and 4 relative cost units (rcu) per metric ton, respectively. The time horizon of interest is 48 h. The unit values for the two products are assumed to be equal at 5 rcu/t. An opportunity of exchanging heat exists between the *Reaction* task which requires cooling, and the *Distillation* task, which requires heating, assuming appropriate temperature levels. An equipment configuration that would permit this heat exchange to be realized is illustrated in Fig. 10.3.

The objective function for the literature example is the maximization of profit, which is defined as the difference between revenue and operating cost. The operating cost consists of consumed external utility costs.

10.4.1 Results and Discussion

The literature example results for the scenario without heat integration are summarized in Table 10.2. The discrete-time model proposed by Papageorgiou et al.

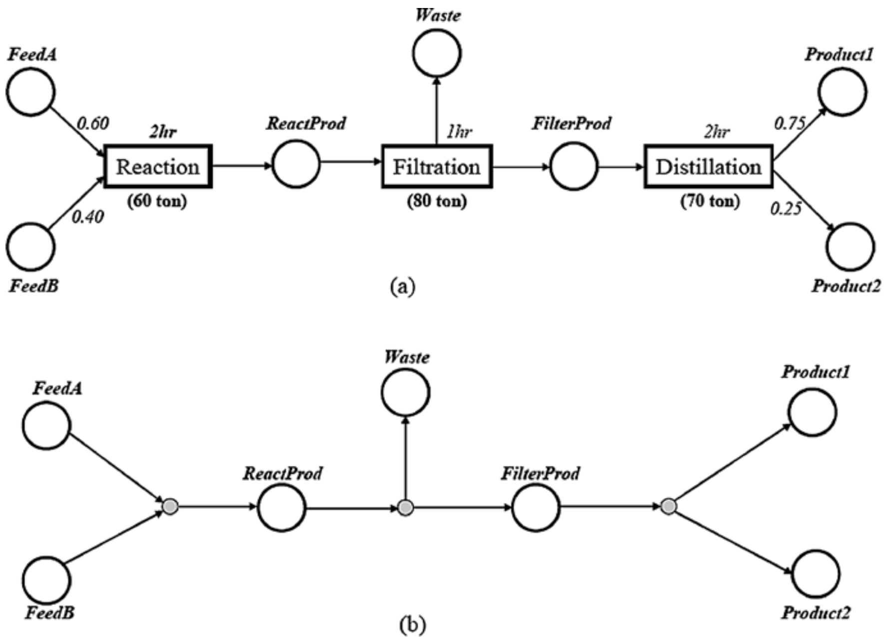


Fig. 10.2 (**a**) State task network (STN) and (**b**) state sequence network (SSN) for the literature example (Papageourgiou et al. (1994))

Table 10.1 Data for the literature example

Task	Recipe		Capacity (t)	Heat Requirements		Duration
Reaction	Feed A	60%	60	Standalone	1,59+0.10 t/h (cooling water)	2
	Feed B	40%		Heat integrated	1.0+0.06 Bt/h (cooling water)	3
Filtration	ReactProd	100%	80	Standalone	0 t/h	1
Distillation	Product1	75%	70	Standalone	0.044+0.0035 Bt/h (steam)	2
	Product2	20%		Heat integrated	0.020+0.0016 Bt/h (steam)	2

(1994) involved 142 binary variables and its solution was based on a 5% margin of optimality in the branch and bound procedure. An integrality gap of 4.76% was observed.

More significantly, a suboptimal objective value of 2944.1 rcu was reported as an optimal solution. Using the uneven discretization of time formulation proposed in this chapter, a globally optimal value of 3081.8 rcu was obtained in 24.5 CPU s. Only 72 binary variables were necessary and the model solution was based on

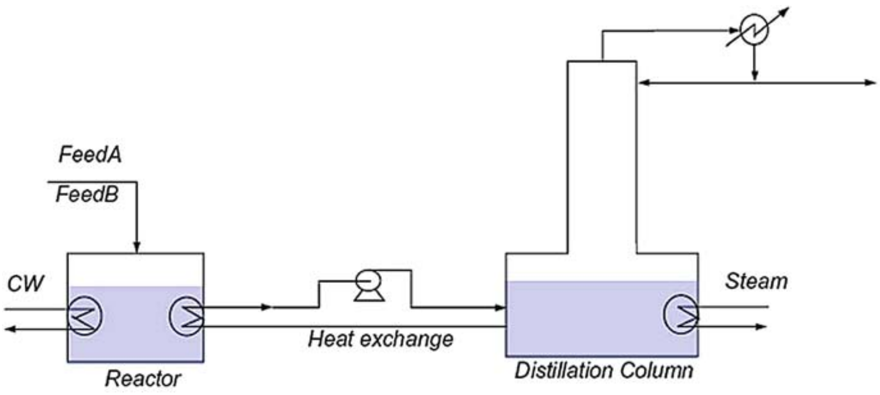


Fig. 10.3 Direct heat exchange network for example process

Table 10.2 Results from the literature example without heat integration

	Papageorgiou et al. (1994)	This formulation
Binary variables	142	72
Objective value (rcu)	2944.1	3081.8
Product1 (t)	945	990
Product2 (t)	315	330
Steam (t)	10.4	10.9
Cooling water (t)	318.8	334.0
Margin of optimality	5%	0%
Integrality gap	4.76%	0%
CPU time (s)	N/A	24.5

0% margin of optimality. An integrality gap of 0% was observed. The Gantt chart corresponding to the globally optimal solution is shown in Fig. 10.4.

Table 10.3 shows the results obtained for the scenario involving heat integration. Both the discrete-time formulation proposed by Papageorgiou et al. (1994) and the uneven discretization of time formulation presented in this chapter gave an objective value of 3644.6 rcu, which is an improvement of 18.3% from the standalone scenario. However, the uneven discretization of time formulation only involves 96 instead of 188 binary variables. As a result the solution of this model was based on 1.3% instead of 5% margin of optimality. A 0% integrality gap was observed. The Gantt chart for the heat-integrated schedule is shown in Fig. 10.5. The *Reaction* task batches that operate in a standalone mode are marked with an *X*, whilst the rest operate in a heat-integrated mode. All the *Distillation* task batches operate in a heat-integrated mode. It is evident from the Gantt chart that the aforementioned condition of common time intervals is obeyed in all the instances that involve heat integration.

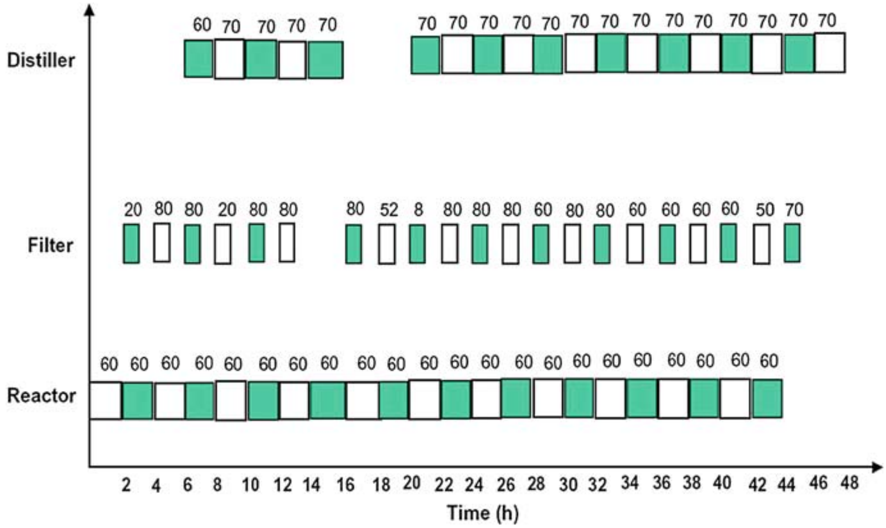


Fig. 10.4 Optimal schedule without heat integration

Table 10.3 Results from the literature example with heat integration

	Papageorgiou et al. (1994)	This formulation
Binary variables	188	96
Objective value (rcu)	3644.6	3644.6
Product1 (t)	720	720
Product2 (t)	240	240
Steam (t)	3.6	3.6
Cooling water (t)	107.2	107.2
Margin of optimality	5%	<1.3%
Integrality gap	0.37%	0%
CPU time (s)	N/A	9812.7

10.5 Industrial Case Study

Figure 10.6 is the flowsheet for the industrial case study used to illustrate the application of the method proposed. Figure 10.7 is the corresponding SSN. Table 10.4 shows the scheduling data. An opportunity for heat integration exists between *Reaction*2 task, which is conducted in *R*3 and *R*4 units, and *Evaporation* task, which is conducted in *EV*1 and *EV*2 units. The cooling load required in *Reaction*2 task is 5 energy units, whilst the heating load required for the *Evaporation* task is 4 energy units. Cooling water and steam are used for cooling and heating and cost 15 and 8 rcu, respectively. Product (state s6) selling price is 100 rcu. The time horizon of interest is 15 h. The capacity of each batch processed in the processing units is 80% of the 10 t design capacity.

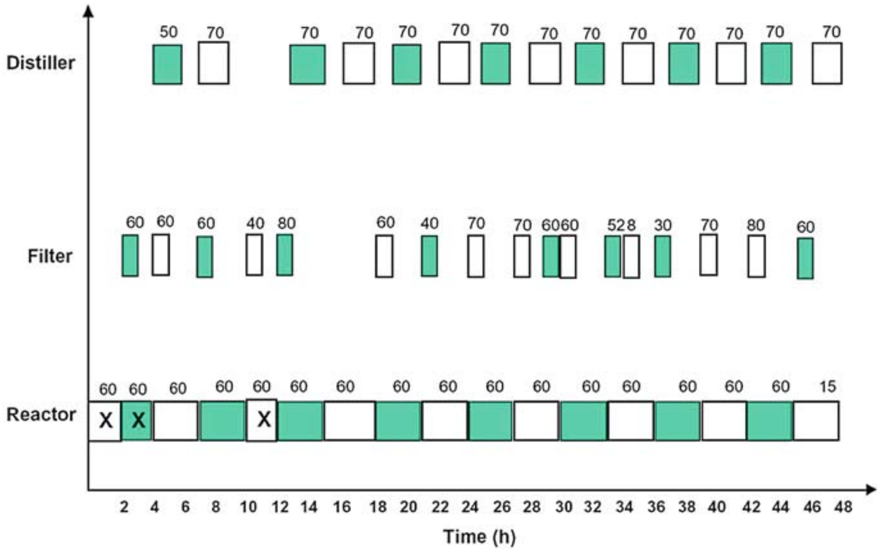


Fig. 10.5 Optimal schedule with heat integration

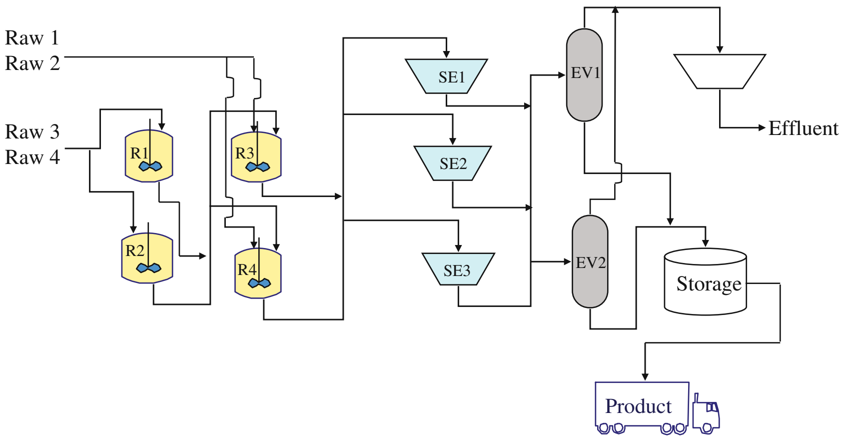


Fig. 10.6 Flowsheet for the industrial case study

10.5.1 Results and Discussion

The case study was solved using the uneven discretization of time formulation presented in this chapter. The mathematical model for the scenario without heat integration (standalone mode) involved 88 binary variables and gave an objective value of 1060 rcu. This value corresponds to the production of 14 t of product and external utility consumption of 12 energy units of steam and 20 energy units

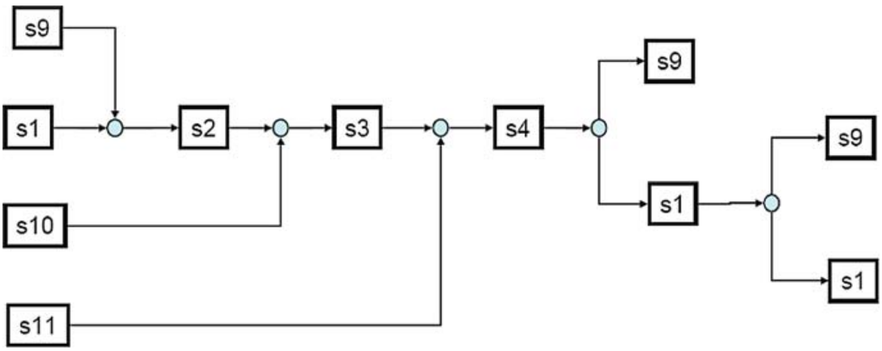


Fig. 10.7 SSN for the industrial case study

Table 10.4 Data for the case study

Unit	*j*	Capacity	Suitability	Mean processing time
R1	1	10	Reaction 1	2
R2	2	10	Reaction 1	2
R3	3	10	Reaction 2, reaction 3	3,1
R4	4	10	Reaction 2, reaction 3	3,1
SE1	5	10	Settling	1
SE2	6	10	Settling	1
SE3	7	10	Settling	1
EV1	8	10	Evaporation	3
EV2	9	10	Evaporation	3
State			Storage capacity	Initial amount
s1			Unlimited	Unlimited
s2			100	0
s3			100	0
s4			100	0
s5			100	0
s6			100	0
s7			100	0
s8			100	0
s9			Unlimited	Unlimited
s10			Unlimited	Unlimited
s11			Unlimited	Unlimited
			Stoichiometric data	
State			Ton/ton output	Ton/ton product
s1			0.20	
s9			0.25	
s10			0.35	
s11			0.20	
s7				0.7
s8				1

of cooling water. On the other hand, the model for the heat-integrated scenario required 120 binary variables and gave an objective value of 1256 rcu, which corresponds to 18.5% improvement in objective value. The product throughput was unaltered, but the external utility consumption was reduced to 0 t for steam and 18 t for cooling water. The absolute elimination of steam requirement is due to the fact that all *Evaporation* task batches operate in a heat-integrated mode. A 0% margin of optimality in the branch and bound procedure was used and a 0% integrality gap observed in both the heat-integrated and standalone scenarios. The results were obtained within 26 CPU s.

The Gantt chart for the heat-integrated schedule is shown in Fig. 10.8. All the *Evaporation* tasks are conducted in a heat-integrated mode. The first batch of the *Evaporation* task in *EV*2 exchanges heat with the third batch of *Reaction*2 task in *R*3, whilst the second batch in *EV*2 exchanges heat with the third batch of *Reaction*2 task in *R*4. Only one batch of the Evaporation task is processed in *EV*1. This batch exchanges heat with the last batch of the *Reaction*2 task in *R*3.

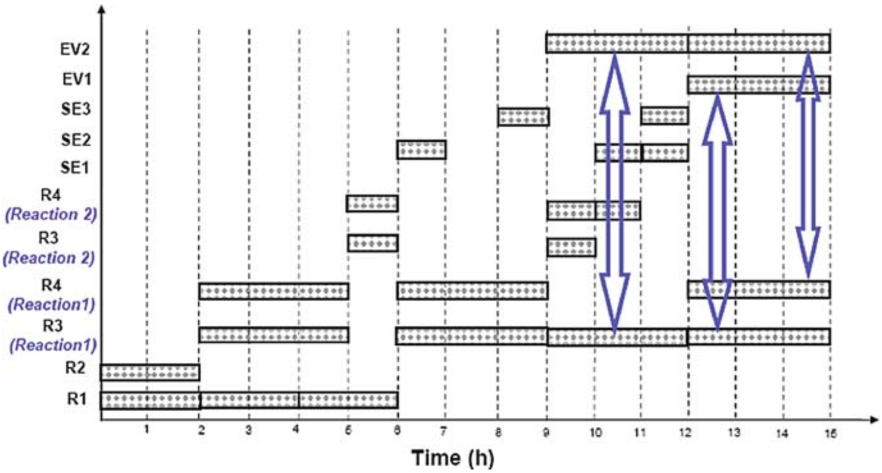


Fig. 10.8 Gantt for the case study – heat integration (Majozi, 2006)

10.6 Conclusions

A uneven discretization of time mathematical formulation for direct heat integration of multipurpose batch plants has been presented. The formulation results in smaller problems compared to the discrete-time formulation, which renders it applicable to large-scale problems. Application of the formulation to an industrial case study showed an 18.5% improvement in objective function for the heat-integrated scenario relative to the standalone scenario.

10.7 Exercise

Task: Revisit the case study shown in Figs. 10.6 and 10.7 and apply the presented formulation over a 20 h time horizon. Assess the benefits of heat integration in this particular case, i.e. compare the energy requirements for the heat integrated case to the standalone case.

References

Adonyi, R., Romero, J., Puigjaner, L., Friedler, F., 2003. Incorporating heat integration in batch process scheduling. Appl. Therm. Eng. 23(14): 1743–1762

Corominas, J., Espuna, A., Puigjaner, L., 1993. A new look at energy integration in multiproduct batch processes. Comput. Chem. Eng. 17(Suppl.): S15–S20

Ivanov, B., Peneva, K., Bancheva, N., 1992. Heat integration of batch vessels at fixed time interval. I. Schemes with recycling main fluids. Hung. J. Ind. Chem., 20: 225–231

Kemp, I.C., Deakin, A.W., 1989. The cascade analysis for energy and process integration of batch processes. Part 1. Calculation of energy targets. Chem. Eng. Res. Des., 67: 495–509

Kemp, I.C., MacDonald, E.K., 1987. Energy and process integration in continuous and batch processes. Inst. Chem. Eng. Symp. Ser. 105: 185–200

Kemp, I.C., MacDonald, E.K., 1988. Application of pinch technology to separation, reaction and batch processes. Inst. Chem. Eng. Symp. Ser., 109: 239–257

Kondili, E., Pantelides, C.C., Sargent, R.W.H., 1993. A general algorithm for short-term scheduling of batch operations. Part I-MILP formulation. Comput. Chem. Eng., 17: 211–227

Linnhoff, B., Ashton, G.J., Obeng, E.D.A., 1988. Process integration of batch processes. Inst. Chem. Eng. Symp. Ser., 109: 221–237

Linnhoff, B., Mason, D.R., Wardle, I., 1979. Understanding heat exchanger networks. Comput. Chem. Eng., 3: 295–302

Majozi, T., 2006. Heat integration of multipurpose batch plants using a continuous-time formulation. Appl. Thermal Eng. J., 26: 1369–1377

Majozi, T., Zhu, X.X., 2001. A novel continuous time MILP formulation for multipurpose batch plants. 1. Short-term scheduling. Ind. Eng. Chem. Res., 40(25): 5935–5949

Obeng, E.D.A., Ashton, G.J., 1988. On pinch technology based procedures for the design of batch processes. Chem. Eng. Res. Des., 6: 255–259

Papageorgiou, L.G., Shah, N., Pantelides, C.C., 1994. Optimal scheduling of heat-integrated multipurpose plants. Ind. Eng. Chem. Res., 33: 3168–3186

Peneva, K., Ivanov, B., Bancheva, N., 1992. Heat integration of batch vessels at fixed time interval. II. Schemes with intermediate heating and cooling agents. Hung. J. Ind. Chem., 20: 233–239

Pinto, T., Novais, A.Q., Barbosa-Póvoa, A.P.F.D., 2003. Optimal design of heat-integrated multipurpose batch facilities with economic savings in utilities: a mixed integer mathematical formulation. Ann. Oper. Res., 120(1-4): 201–230

Umeda, T., Harada, T., Shiroko, K., 1979. A thermodynamic approach to the synthesis of heat integration systems in chemical processes. Comput. Chem. Eng., 3: 273–282

Vaklieva-Bancheva, N., Ivanov, B.B., Shah, N., Pantelides, 1996. C.C., Heat exchanger network design for multipurpose batch plants. Comput. Chem. Eng. 20(8): 989–1001

Vaselenak, J.A., Grossmann, I.E., Westerberg, A.W., 1986. Heat integration in batch processing. Ind. Eng. Chem. Process Des. Dev. 25: 357–366

Chapter 11
Heat Integration in Multipurpose Batch Plants: II. Indirect Heat Integration

Overview This chapter is, in essence, an extention of Chapter 10 which focused on direct heat integration as encountered in the absence of heat storage (Majozi, J. Cleaner Prod., 17: 945–950, 2006). The inclusion of heat storage in the exploration of energy saving opportunities through heat integration adds more degrees of freedom in the analysis, which is likely to improve the optimum point. The necessity of heat storage arises from the time dimension that is inherent in batch plants as mentioned severally in the foregoing chapters. Storage of heat allows the time dimension to be bypassed by allowing the task that serves as the heat source to take place before the task that is appropriate to serve as the corresponding heat sink without losing the opportunity for heat integration. Heat transfer fluids, e.g. water, are normally used for this purpose. The presented mathematical formulation exhibits an MILP structure for the fixed capacity of storage. Application of the proposed method to an agrochemical facility demonstrates savings of more than 75% in external utility steam consumption.

11.1 Problem Statement

The problem addressed in this chapter can be stated as follows.
Given:

(i) production scheduling data, i.e. equipment capacities, task durations, time horizon of interest, recipe for each product as well as cost of raw materials and selling price of final products,
(ii) hot and cold duties for tasks that require heating and cooling, respectively,
(iii) cost of cooling water and steam,
(iv) operating temperatures for the heat source and the heat sink operations,
(v) allowed minimum temperature difference and
(vi) available heat storage capacity

determine the production schedule that results in minimum energy use or maximum profit. In the context of this chapter, profit is defined as the difference between

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_11,

revenue and operating costs. The latter constitute raw material costs and external utility (cooling water and steam) costs. It is assumed that sufficient temperature driving forces exist between matched tasks for process–process heat transfer. Also, each task is allowed to operate either in an integrated or standalone mode. The integrated mode, in the context of this chapter, is twofold, since a unit is allowed to be integrated with either heat storage or another operating unit. If heat integration cannot supply sufficient duty, external utility is supplied to complement the deficit. Whilst direct heat integration requires involved tasks to be active within a common time interval to effect direct heat transfer, they need not necessarily commence nor end at the same time. Moreover, the heat integrated tasks can either belong to the same process or distinct processes within reasonable proximity.

11.2 Mathematical Model

As aforementioned, the mathematical model proposed in this chapter is an extension of the mathematical formulation presented in Chapter 10. It is based on the uneven discretization of the time horizon as shown in Fig. 11.1 and entails the following sets, variables and parameters.

Sets

$U = \{u | u \text{ is a heat storage unit}\}$
$J = \{j | j \text{ is a processing unit}\}$
$J_c = \{j_c | j_c \text{ is a processing unit that requires cooling}\} \subset J$
$J_h = \{j_h | j_h \text{ is a processing unit that requires heating}\} \subset J$
$P = \{p | p \text{ is a time point}\}$
$S_{\text{in},j} = \left\{s_{\text{in},j} | s_{\text{in},j} \text{ is an input stream to a processing unit}\right\}$
$S_{\text{out},j} = \left\{s_{\text{out},j} | s_{\text{out},j} \text{ is an output stream froma processing unit}\right\}$

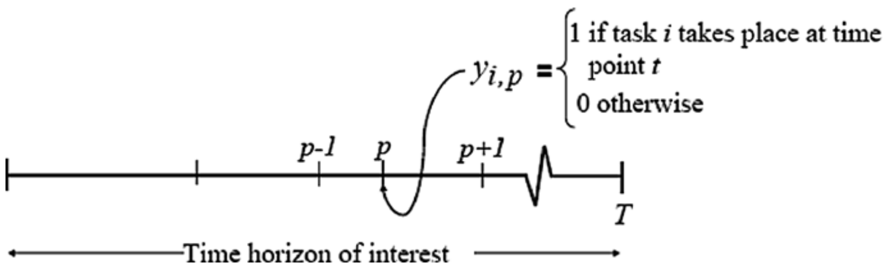


Fig. 11.1 Continuous time representation of the time horizon (Majozi, 2009)

Variables

$t_p\left(s_{\text{out},\,j},p\right)$ = time at which the stream is produced from unit j
$t_u\left(s_{\text{in},\,j},p\right)$ = time at which the stream enters the processing unit j
$T_0\,(u,p)$ = initial temperature in the storage vessel at time point p
$T_f\,(u,p)$ = final temperature in the storage vessel at time point p
$CW\,(j,p)$ = amount of external cooling required by operation j at time point p
$ST\,(j,p)$ = amount of external heating required by operation j at time point p
$Q\,(j,u,p)$ = amount of heat exchanged with storage at time point p

$$y\,(j,u,p) = \begin{cases} 1 & \leftarrow \text{if unit } j \text{ is exchanging heat with storage unit } u \\ 0 & \leftarrow \text{otherwise} \end{cases}$$

$$x\left(j,j',p\right) = \begin{cases} 1 & \leftarrow \text{if unit } j \text{ is exchanging heat with another unit } j' \\ 0 & \leftarrow \text{otherwise} \end{cases}$$

$$y\left(s_{\text{in},\,j},p\right) = \begin{cases} 1 & \leftarrow \text{if unit } j \text{ is active at time point } p \\ 0 & \leftarrow \text{otherwise} \end{cases}$$

Parameters

$T\,(j)$ = operating temperature for processing unit j
$\tau\,(j)$ = duration of operation j in standalone mode
$\tau'\left(j,j'\right)$ = duration of operation j when directly heat integrated
$\tau''\,(j,u)$ = duration of operation j when integrated with storage
ΔT^L = minimum temperature difference
$Q\,(j)$ = amount of heat required by or removed from the operating unit j
$M\,(u)$ = capacity of heat storage u
T_{start} = temperature of storage at the beginning of the time horizon

Constraints

The mathematical model is based on the superstructure shown in Fig. 11.2. The heat transfer fluid in heat storage remains in the storage vessel during heat transfer with only the process fluid pumped around. The superstructure also shows that each unit is capable of receiving external heating or cooling in addition to direct and indirect heat integration.

In addition to scheduling constraints that have been presented in detail in Chapter 2, the following constraints are necessary to cater for heat storage. Constraints (11.1) and (11.2) ensure that direct heat integration involves exactly one pair of units so as to simplify process operability. In essence, these constraints state that if 2 units are heat integrated at any given point in time, then these units must also be active at that point in time. However, if a unit is active at a given time point it is not necessary that it be heat integrated with another unit.

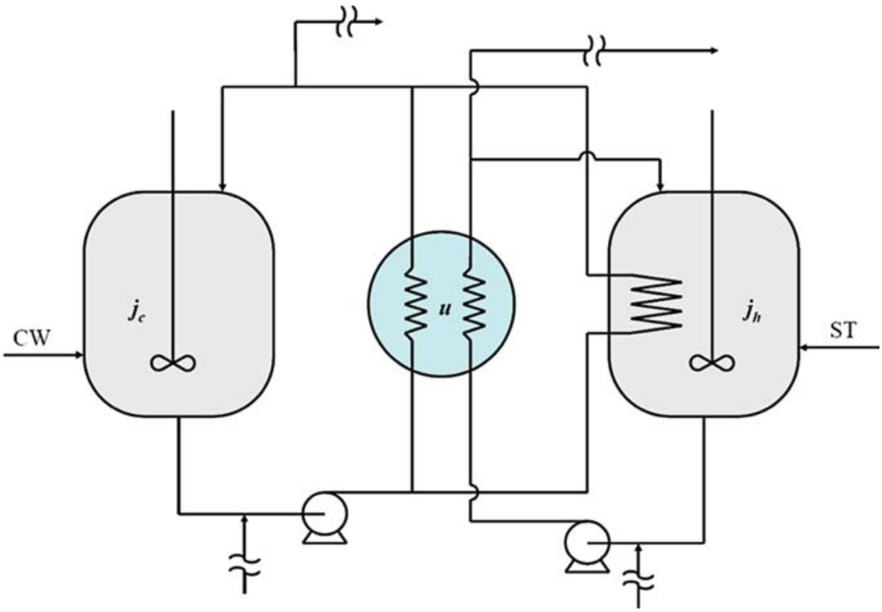


Fig. 11.2 Superstructure for the mathematical model (Majozi, 2009)

$$\sum_{j' \in J_c} x\left(j, j', p\right) \leq y\left(s_{\text{in},\,j}, p\right), \ \forall p \in P, j \in J_h, s_{\text{in},\,j} \in S_{\text{in},\,j} \tag{11.1}$$

$$\sum_{j \in J_h} x\left(j, j', p\right) \leq y\left(s_{\text{in},\,j'}, p\right), \ \forall p \in P, j \in J_c, \ s_{\text{in},\,j'} \in S_{\text{in},\,j} \tag{11.2}$$

Constraints (11.3) stipulates the quantity of heat transferred to storage from a hot unit at the beginning of the time horizon. Constraints (11.4) and (11.5) quantify the amount of heat transferred and received from storage unit, respectively. They ensure that if there is no heat integration between a processing unit and storage, then the amount of heat related to storage is not disturbed.

$$Q\left(j, u, p0\right) = M\left(u\right) c_p \left(T_f\left(u, p1\right) - T_{\text{Start}}\right) y\left(j, u, p0\right), \quad \forall j \in J_c \subset J, u \in U \tag{11.3}$$

$$Q\left(j, u, p-1\right) = M\left(u\right) c_p \left(T_f\left(u, p\right) - T_0\left(u, p-1\right)\right) y\left(j, u, p-1\right), \quad \forall j \in J_c \subset J, p \in P, p > p0, u \in U \tag{11.4}$$

$$Q\left(j', u, p-1\right) = M\left(u\right) c_p \left(T_0\left(u, p-1\right) - T_f\left(u, p\right)\right) y\left(j', u, p-1\right), \quad \forall j \in J_h \subset J, p \in P, p > p0, u \in U \tag{11.5}$$

Constraints (11.6) ensures that only one unit is heat integrated with storage at any given point in time. Constraints (11.7) and (11.8) ensure that the temperature of the storage unit is not changed if there is no heat integration with any unit. These constraints carry the same meaning as constraints (11.3)–(11.5). Nonetheless, they are necessary since they pertain to temperature whilst the latter pertain to the amount of heat. Overall, constraints (11.3)–(11.8) govern the relationship between heat and temperature of storage.

$$\sum_{j \in J_c} y(j,u,p) + \sum_{j' \in J_h} y\left(j',u,p\right) \le 1, \quad \forall p \in P, u \in U \tag{11.6}$$

$$T_0(u,p-1) \le T_f(u,p) + \max_j \{T(j)\} \left(\sum_{j \in J_c} y(j,u,p-1) + \sum_{j' \in J_h} y\left(j',u,p-1\right) \right), \quad \forall p \in P, p > p0, u \in U \tag{11.7}$$

$$T_0(u,p-1) \ge T_f(u,p) - \max_j \{T(j)\} \left(\sum_{j \in J_c} y(j,u,p-1) + \sum_{j' \in J_h} y\left(j',u,p-1\right) \right), \quad \forall p \in P, p > p0, u \in U \tag{11.8}$$

Constraints (11.9) ensures that the initial temperature in heat storage at any given point in time is the same as the final temperature at the last time point. This condition is always true, regardless of the heat integration status in the previous time point.

$$T_0(u,p) = T_f(u,p-1), \forall p \in P, u \in U \tag{11.9}$$

Constraints (11.10) and (11.11) ensure that if there is heat integration between any unit and heat storage, then the stipulated minimum driving force should be obeyed. Constraints (11.10) applies if heat storage is integrated with the heat source, whilst constraints (11.11) applies if heat storage is integrated with the heat sink.

$$T(j) - T_f(u,p) \ge \Delta T^L - \max_j \{T(j)\} (1 - y(j,u,p-1)), \quad \forall p \in P, p > p0, j \in J_c \subset J, u \in U \tag{11.10}$$

$$T_f(u,p) - T(j) \ge \Delta T^L - \max_j \{T(j)\} (1 - y(j,u,p-1)), \quad \forall p \in P, p > p0, j \in J_h \subset J, u \in U \tag{11.11}$$

Constraints (11.12) states that cooling in any heat source will be accomplished either by direct heat integration, external cooling or heat integration with storage. Constraints (11.13) is similar to constraints (11.12) but applies to a heat sink.

$$Q(j)\, y\left(s_{\mathrm{in},j},p\right) = Q(j,u,p) + CW(j,p) + \sum_{j' \in J_h} \min_{j,j'} \left\{Q(j), Q\left(j'\right)\right\} x\left(j,j,'p\right) \quad \forall j \in J_c \subset J, p \in P, u \in U \tag{11.12}$$

$$Q(j)\, y\left(s_{\mathrm{in},j},p\right) = Q(j,u,p) + ST(j,p) + \sum_{j \in J_c} \min_{j,j'} \left\{Q(j), Q\left(j'\right)\right\} x\left(j,j,'p\right) \quad \forall j \in J_h \subset J, p \in P, u \in U \tag{11.13}$$

Constraints (11.14) and (11.15) state that if a unit is directly heat integrated with another unit, then it cannot be simultaneously integrated with heat storage. This is also a condition imposed solely to simplify operability of the overall process.

$$\sum_{j' \in J_h} x\left(j,j,'p\right) + y\left(j,u,p\right) \leq 1, \forall j \in J_c \subset J, p \in P, u \in U \tag{11.14}$$

$$\sum_{j \in J_c} x\left(j,j,'p\right) + y\left(j',u,p\right) \leq 1, \forall j' \in J_h \subset J, p \in P, u \in U \tag{11.15}$$

Constraints (11.16) is a feasibility constraints which ensures that if a unit is not integrated with storage, then the associated duty should not exist.

$$\delta y\left(j,u,p\right) \leq Q\left(i,u,p\right) \leq \max_{j \in J}\left(Q\left(j\right), Q\left(j'\right)\right) y\left(j,u,p\right) \tag{11.16}$$

Constraints (11.17) shows how the variation in duration due to the heat integration mode is accounted for in the mathematical model. It is very likely that the duration times will be affected by the mode of operation and this should not be ignored in the formulation.

$$\begin{aligned} t_p\left(s_{\text{out},\,j},p\right) = t_u\left(s_{\text{in},j},p-1\right) + \tau\left(j\right)\left(1 - y\left(j,j,'p\right) - y\left(j,u,p\right)\right) \\ +\tau''\left(j,u\right) y\left(j,u,p\right) + \tau'\left(j,j'\right) y\left(j,j,'p\right), \\ \forall j \in J, p \in P, u \in U, s_{\text{in},\,j}, s_{\text{out},j} \in S \end{aligned} \tag{11.17}$$

The foregoing constraints constitute the full heat storage model. With the exception of constraints (11.3)–(11.5), all the constraints are linear. Constraints (11.3)–(11.5) entail nonconvex bilinear terms which render the overall model a nonconvex MINLP. However, the type of bilinearity exhibited by these constraints can be readily removed without compromising the accuracy of the model using the so called Glover transformation, which has been used extensively in the foregoing chapters of this book. This is demonstrated underneath using constraints (11.3).

$$\begin{aligned} Q\left(j,u,p\right) = M\left(u\right) c_p\left(\Gamma_f\left(j,u,p\right) - \Gamma_0\left(j,u,p\right)\right), \\ \forall j \in J_c \subset J,\, p \in P, p > p0, u \in U \end{aligned} \tag{11.3'}$$

$$\begin{aligned} T_f\left(u,p\right) - T_f^U\left(u\right)\left(1 - y\left(j,u,p-1\right)\right) \leq \Gamma_f\left(j,u,p\right) \\ \leq T_f\left(u,p\right) + T_f^L\left(u\right)\left(1 - y\left(j,u,p-1\right)\right), \\ \forall j \in J_c \subset J, p \in P, p > p0, u \in U \end{aligned} \tag{11.18}$$

$$\begin{aligned} T_f^L\left(u\right) y\left(j,u,p-1\right) \leq \Gamma_f\left(j,u,p\right) \leq T_f^U y\left(j,u,p-1\right), \\ \forall j \in J_c \subset J, p \in P, p > p0, u \in U \end{aligned} \tag{11.19}$$

$$\begin{aligned} T_0\left(u,p-1\right) - T_0^U\left(u\right)\left(1 - y\left(j,u,p-1\right)\right) \leq \Gamma_0\left(j,u,p\right) \\ \leq T_0\left(u,p-1\right) + T_0^L\left(u\right)\left(1 - y\left(j,u,p-1\right)\right), \\ \forall j \in J_c \subset J, p \in P, p > p0, u \in U \end{aligned} \tag{11.20}$$

$$T_0^L(u)\,y(j,u,p-1) \le \Gamma_0(j,u,p) \le T_0^U y(j,u,p-1),\quad \forall j \in J_c \subset J, p \in P, p > p0, u \in U \tag{11.21}$$

Constraints (11.18), (11.19), (11.20) and (11.21) constitute the linearized version of constraints (11.3). The advantage of this linearization technique is that it is exact, which implies that global optimality is assured. The disadvantage, however, is that it requires the introduction of new variables and additional constraints. Consequently, the size of the model is increased. A similar type of linearization is also necessary for constraints (11.4) in order to have an overall MILP model which can be solved exactly to yield a globally optimal solution.

11.3 Case Study

Figure 11.3 is the representation of the case study that is used to demonstrate the performance of the proposed model it is taken from directly from Chapter 10. To facilitate understanding, this case study is described in some detail in this chapter. The plant, which constitutes 30% of production and consumes 55% of utility steam in the multinational agrochemical facility of choice, involves the manufacture of an herbicide. The saturated steam is produced from a coal fired boiler at 10 bar absolute pressure and 3 t/h, although it is only used at 4 bar in the chosen process. The process entails 3 consecutive chemical reactions which take place in 4 reactors. The first reaction, which uses water as a solvent, takes place in reactors R1 and R2.

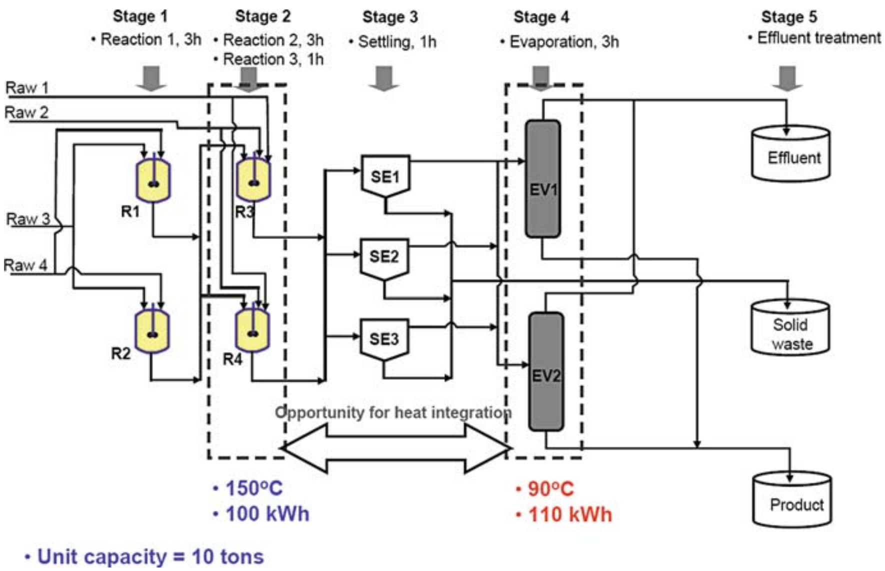


Fig. 11.3 Process flowsheet for the case study (Majozi, 2009)

The 2 raw materials shown in the flowsheet are added manually, almost simultaneously, through the chutes. The intermediate from this reaction is then transferred to reactors R3 and R4, wherein 2 consecutive reactions take place. The first of these 2 reactions is highly exothermic and requires almost 9 t of cooling water (equivalent to 100 kWh). For operational purposes, these 2 consecutive reactions take place in a single reactor. However, some of the intermediate from the first of these 2 reactions can be stored in an intermediate buffer tank prior to the final reaction to improve throughput. Both the second and the third reactions form sodium chloride (NaCl) as a byproduct. Consequently, the intermediate from the third reaction is transferred to settlers SE1, SE2 and SE3, to separate NaCl from the supernatant aqueous solution containing the active ingredient. The salt-free solution is then transferred to evaporators where steam (equivalent to 110 kWh) is used to remove excess water from the product. This excess water is finally dispensed with as effluent, whilst the final product is collected in storage tanks before final formulation, packaging and transportation to customers. All unit operations in this process are at almost atmospheric pressure. The task durations are shown in the flowsheet, Fig. 11.3. Each of the units has a capacity of 10 t. The additional scheduling data appears in Table 10.4.

The temperatures for the exothermic second reaction (150°C) and endothermic evaporation in stage 4 (90°C), allow for possible heat integration. Consequently, the foregoing mathematical model was applied to this case study. The capacity of storage was fixed at 2 t and the chosen initial temperature was 80°C. It is should be mentioned at this stage that the choice of storage and initial temperature will always have an influence on optimality of energy use. Therefore, these should be optimization variables, instead of parameters. However, this will be subject of another publication. The other relevant information pertaining to the case study is shown in Table 11.1. The objective of the case study is to maximise profit which, in this case, is the difference between revenue and utility costs.

Table 11.1 Data for the case study

Specific heat capacity of water c_p (kJ/kg °C)	4.2
Latent heat of vaporization λ_v (kJ/kg)	2.13×10^6
Product cost (cu/t)	10,000
Steam cost (cu/kWh)	20
Cooling water cost (cu/kWh)	8

11.4 Results

Whilst direct heat integration, i.e. without any use of heat store, resulted in 25% improvement in terms of external cold utility requirements, use of heat storage showed more than 75% improvement. In this particular case study, both direct and

Table 11.2 Summary of results from the case study

	No heat integration	Direct heat integration	Heat storage
Performance index (cost units)[a]	130 200	137 000	138 600
External cold duty (kW)	400	300	100
External hot duty (kW)	330	30	30
CPU time (s)[b]	182.73	26.8	2805.2
Constraints			4144
Continuous variables			962
Binary variables			156

[a]Performance index = revenue – utility costs
[b]Results obtained from GAMS/CPLEXPAR 7.0 using Intel Core 2 Duo 1.2 GHz processor

indirect heat integration resulted in 90% reduction in external hot duty requirements. These results are summarised in Table 11.2. The Gantt charts corresponding to the optimal solutions for standalone operation, direct heat integration and heat integration with storage are shown in Figs. 11.4, 11.5 and 11.6. The arrows in Figs. 11.5 and 11.6 show opportunities for direct and indirect heat integration. In Fig. 11.6, storage is used as heat sink by reactor R4 during time period 2–5 h and also by reactor R3 during time period 6–9 h. the other 3 process integration options involve direct heat integration. This Gantt chart is strictly a consequence of the choice of initial storage temperature and storage capacity. Figure 11.7 shows the temperature profile in the storage unit over the time horizon of interest.

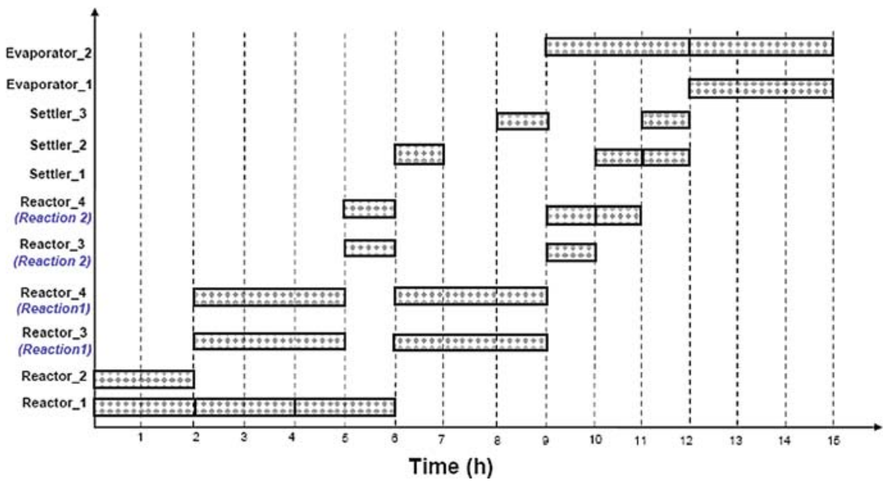


Fig. 11.4 Gantt chart for the case study without heat integration (Majozi, 2009)

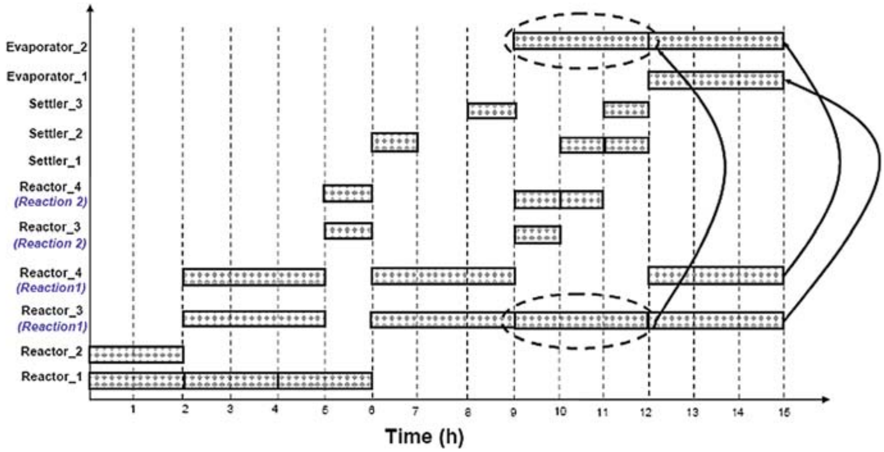


Fig. 11.5 Gantt chart for the case study with direct heat integration (Majozi, 2009)

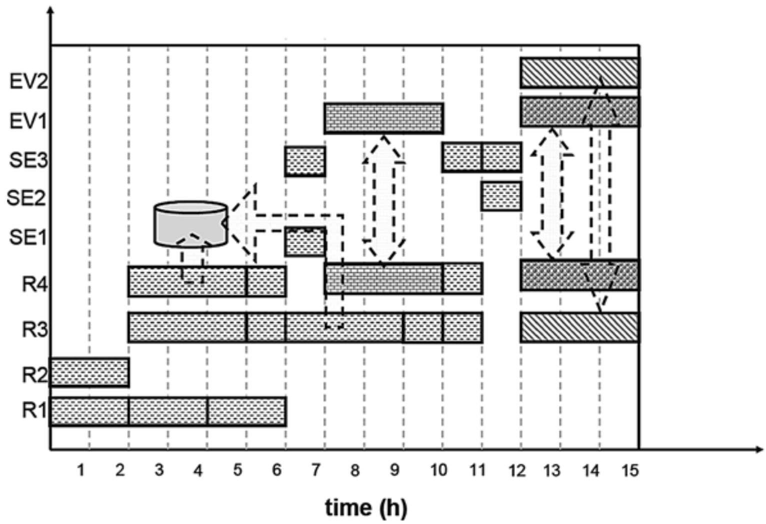


Fig. 11.6 Gantt chart for the case study with indirect heat storage

11.5 Conclusions

A mathematical approach for optimization of energy use in heat integrated multipurpose batch plants has been presented and tested in a case study. The results have shown that heat integration with heat storage considerations can result in energy savings of more than 75%, compared to standalone operation that relies solely on

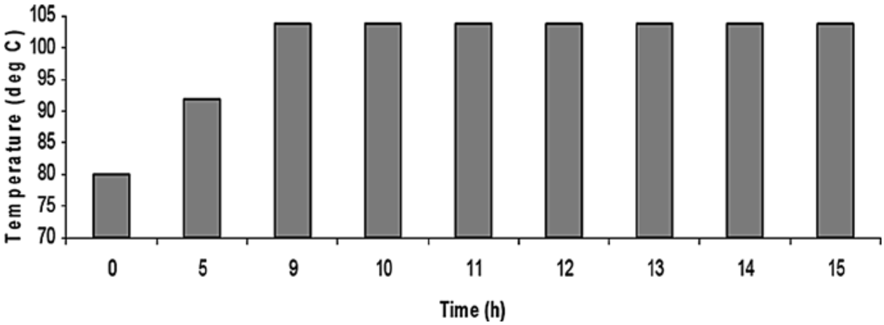


Fig. 11.7 Temperature profile of heat storage

Table 11.3 Data for the case study

Specific heat capacity of water c_p (kJ/kg °C)	4.2
Latent heat of vaporization λ_v (kJ/kg)	2.13×10^6
Product cost (cu/t)	*8,000*
Steam cost (cu/kWh)	*15*
Cooling water cost (cu/kWh)	*5*

external utilities. Moreover, the option of heat storage improves the flexibility of the overall process, since the heat source and the heat sink need not take place at over a common time interval.

11.6 Exercise

Task: Consider the case study presented above, but with revised cost indices as shown in Table 11.3 below and repeat the analysis over the same time horizon.

References

Majozi, T., 2006. Heat integration of multipurpose batch plants using a continuous-time framework. Appl. Thermal Eng., 26: 1369–1377.

Majozi, T., 2009. Minimization of energy use in multipurpose batch plants using heat storage: an aspect of cleaner production. J. Cleaner Prod., 17: 945–950.

Chapter 12
A Graphical Technique for Wastewater Minimisation in Batch Processes

Overview All of the methods presented in the foregoing chapters of this book are based solely on mathematical modelling with only a mention of the availability of graphical techniques in Chapter 1. This is deliberately aimed at presenting the full strength of mathematical techniques before their graphical counterparts. The major limitation of graphical techniques lies mainly in their confinement to 2 dimensions. In situations where problems are characterised by several degrees of freedom, graphical techniques tend to be limited in their applicability. However, batch processes generally embed many degrees of freedom, which render them less amenable to graphical techniques, particularly where accuracy is of essence. Presented in this chapter is a graphical technique for freshwater and wastewater minimisation in completely batch operations. Like in the methods presented in earlier chapters, water minimisation is achieved through exploration of water reuse and recycle opportunities. In the context of this chapter, a completely batch operation is the one in which water reuse or recycle can only be effected either at the start or the end of the process. Before the operation is complete, there exists no opportunity for recycle and reuse. Two instances are considered in the analysis and optimisation procedure. In the first instance, time dimension is taken as a primary constraint and concentration as a secondary constraint. Subsequently, the priority of constraints is reversed so as to demonstrate the effect of the targeting procedure on the final design. Attention is brought to the fact that first and cyclic-state targeting are essential in completely batch operations. Moreover, the exploration and use of inherent storage in batch processes is demonstrated using a real-life case study. Like most graphical techniques, the presented methodology is limited to single contaminants. Moreover, in all the presented examples time is treated as a parameter rather than a variable solely to allow for graphical analysis. Finally, a brief comparison between the results obtained using graphical analysis and results obtained using mathematical modelling is given.

12.1 Introduction

The last two decades have been characterised by intensified research in the area of mass integration in chemical processes. This is a direct result of the ever-tightening

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3_12,

environmental constraints as well as increased global awareness on sustainable development, hence cleaner production. However, most of the developments have been targeted at continuous processes at steady-state (Takama et al., 1979; El-Halwagi and Manousiouthakis, 1990; Wang and Smith, 1994, 1995; Kiperstok and Sharratt, 1995; Olesen and Polley, 1997; Jödicke et al., 2000; Hallale, 2002). Equal developments in batch chemical processes are still at their early stages for several reasons. Firstly, most process integration techniques for continuous processes are confined in 2 dimensions, which allow for graphical analysis. It is generally understood that techniques that are solely based on graphical analysis would have very limited benefits in batch processes due to the intrinsic time dimension. In essence, it is mainly for this reason that most process integration methodologies in batch operations are based on mathematical modelling, which is virtually dimensionally unconstrained (Vaselanak et al., 1986; Grau et al., 1996; Sanmartí et al., 1998; Yao and Yuan, 2000; Gouws et al., 2008; Gouws and Majozi, 2009). Secondly, the incorporation of process integration in batch processes is perceived to lead to reduced flexibility, which is the main feature of batch processes. Thirdly, there is a general understanding that batch processes have intrinsic variations which eventually lead to deviations from predetermined schedules. This would make targeting extremely difficult, if not impossible. Fourthly, it has always been assumed that process integration would not have much significance in batch operations, since energy and water usually constitute a small component of operating costs (Obeng and Ashton, 1988). Stringent environmental regulations are steadily rendering this notion untrue.

Foo et al. (2004) developed a methodology for the synthesis of mass exchange network in batch processes with a focus on utility or mass separating agent (MSA) targeting. This methodology is an adaptation of the work by El-Halwagi and Manousiouthakis (1989) for mass exchanger network synthesis in continuous processes, combined with the cascade analysis for batch heat integration developed by

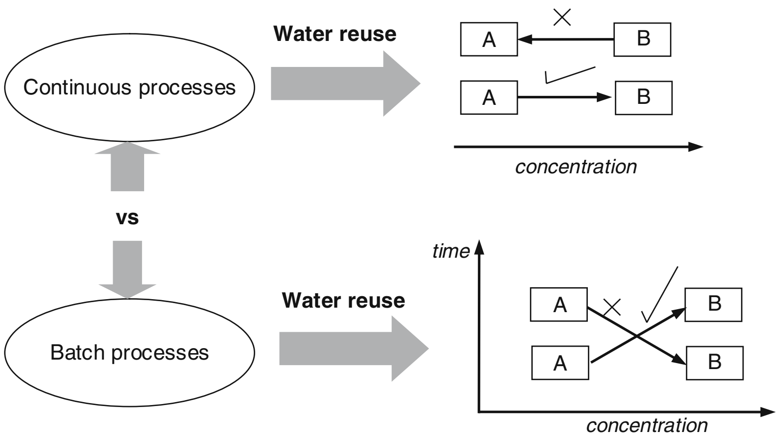


Fig. 12.1 Comparison of continuous and batch processes (Majozi, 2005)

Kemp and Deakin (1989). A new targeting tool called time-dependent composition interval table that facilitates vertical cascading of mass exchange between rich and lean streams through composition intervals forms the basis of this methodology. Like similar techniques, the schedule is considered known beforehand, thereby rendering time a parameter instead of a variable. However, as emphasised repeatedly in the foregoing chapters, in order to handle batch operations effectively, the time dimension cannot be ignored due to the fact that almost all operations within the batch process environment are time dependent. This is illustrated in Fig. 12.1, which is explained in detail in Chapter 4 (see Fig. 4.4).

In 1995, Wang and Smith developed a graphical analysis technique for water minimisation in batch processes, which forms the basis for the argument put forth in this chapter. A detailed presentation of this technique is, therefore, necessary in order to facilitate understanding of the subsequent sections of this chapter.

12.1.1 Wang and Smith Approach for Water Minimisation in Batch Processes

To facilitate understanding of their methodology as applied to setting water demand and wastewater generation targets, Wang and Smith (1995) formulated an example comprising of three processes as shown in Table 12.1. It should be noted that these data are limiting, i.e. the inlet and outlet concentrations of contaminant have been set to the maximum values. Ignoring time constraints and solving the problem as a continuous process yields a water demand target of 155 t/h. However, the fact that this is a system of batch operations means that the inherent time constraints cannot be ignored.

Since mass transfer takes place within a limited period of time, which is not the case with continuous processes, the amount of contaminant transferred is measured in kg instead of kg h^{-1}. The amount of contaminant is calculated as:

$$\Delta m = f \Delta C \Delta t \tag{12.1}$$

where ΔC is the change in concentration, Δt is the duration of mass transfer and f is the water flowrate required. To understand the role of the time dimension better, constraints (12.1) is rewritten as:

Table 12.1 Data for the Wang and Smith (1995) example

		Concentrations (ppm)		Time (h)	
Process number	Flowrate (t/h)	$C_{in,max}$	$C_{out,max}$	t_0	t_1
1	100	100	400	0.5	1.5
2	80	0	200	0	0.5
3	50	100	200	0.5	1.0

$$\frac{\Delta m}{\Delta C} = f\Delta t \tag{12.2}$$

In constraints (12.2), $f\Delta t$ is the amount of water required to remove Δm (kg) of contaminant associated with a ΔC change in concentration. If ΔG is assigned to the amount of water required, the following constraints is obtained:

$$\Delta G = f\Delta t \tag{12.3}$$

Plotting ΔG against time as shown in Fig. 12.2 clearly represents the behaviour of a batch water-using operation.

To ensure that both concentration and time constraints are met, this analysis should be applied in each of the concentration intervals. It would be expected that a similar approach to the Problem Table Algorithm (Linnhoff and Flower, 1978) should be applied in order to ensure that a specific minimum concentration difference holds in each of the intervals. This would entail shifting of the inlet and outlet concentrations for the process streams as it was done in setting the energy targets. However, the fact that the limiting concentration constraints have been built into the problem makes the shifting of concentrations irrelevant. The concentration intervals are demarcated by the inlet and outlet concentrations as shown in Fig. 12.3.

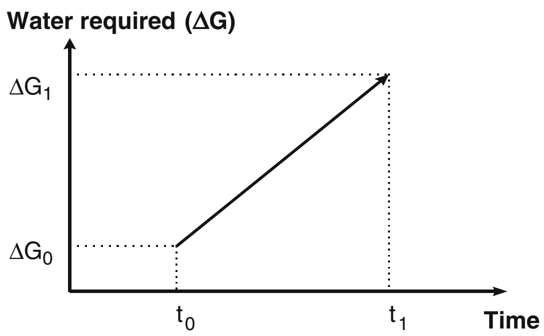


Fig. 12.2 A batch water-using process on water required versus time

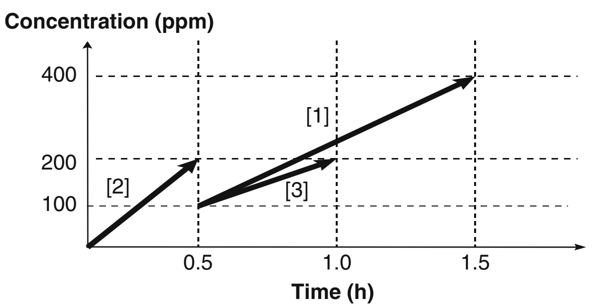


Fig. 12.3 Demarcation of concentration intervals

Targeting is performed by cascading water from one concentration interval to the next, until the last concentration interval. Within each concentration interval, water is cascaded from one time subinterval to the next without degradation. This appears to be the mass transfer version of the targeting procedure presented for heat exchangers.

In concentration interval (0–100 ppm) only process 2 exists, Fig. 12.3. Targeting in this concentration interval is shown in Fig. 12.4. The amount of water required is 40 t, which leaves the process at a concentration of 100 ppm.

In the concentration interval (100–200 ppm), processes 1, 2 and 3 coexist as depicted in Fig. 12.3. The total amount of water required is:

$$[100\,(1.5-0.5)]_{\text{process 1}} + [80\,(0.5-0)]_{\text{process 2}} + [50\,(1.0-0.5)]_{\text{process 3}} = 165\,\text{t}$$

This water should be supplied at a concentration of 100 ppm. Targeting in this concentration interval is shown in Fig. 12.5.

However, only 40 t of water is available from the first concentration interval. This implies that the deficit should be supplied by fresh water. The amount of fresh water required is calculated as:

$$\frac{(165-40)\,(200-100)}{(200-0)} = 62.5\ t \qquad (12.5)$$

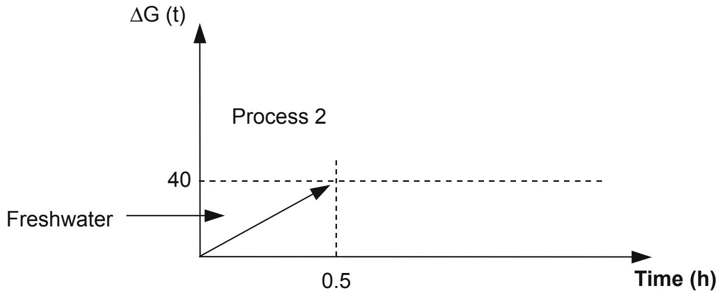


Fig. 12.4 Targeting in concentration interval (0 → 100 ppm)

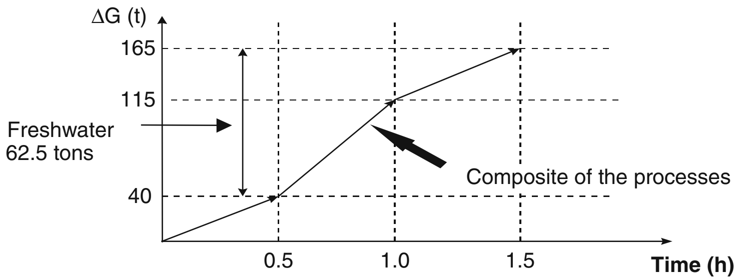


Fig. 12.5 Targeting in concentration interval (100 → 200 ppm)

Therefore, the total amount of contaminated water is now 102.5 t at a concentration of 200 ppm. This water can be reused in the next concentration interval.

In concentration interval (200–400 ppm), only process 1 exists, Fig. 12.3. Targeting in this concentration interval is shown in Fig. 12.6. The total amount of water required is 100 t at a concentration of 200 ppm. Since 102.5 t of water at a concentration of 200 ppm is available for reuse from the previous interval, no fresh water is required in this interval. This eventually sets 200 ppm as the pinch concentration, i.e. the concentration beyond which the water target does not change. The total amount of wastewater generated from processes is 102.5 t, which is the minimum wastewater target.

As it can be inferred from Fig. 12.6, in the time period (0–0.5 h), 40 t of water is available for reuse. In the time period (0.5–1.0 h), 50 t of water is required, but there is only 37.5 t of water available. Therefore, 12.5 t of water is required from storage. In the time period (1.0–1.5 h), 50 t of water is required and only 25 t is available, thus 25 t of water is required from storage. This implies that 37.5 t of water from time period (0–0.5 h) should be stored for reuse in the time periods (0.5–1.0 h) and (1.0–1.5 h). Figure 12.7 shows the design to meet the target.

Implicit in their analysis (Wang and Smith, 1995), were a number of assumptions which were not immediately obvious. These were surfaced when this methodology had to be applied to a set of completely batch processes in an agrochemical facility.

Firstly, the analysis implicitly allows water reuse even in a situation where the source and sink processes are simultaneously active as observed in Fig. 12.6. According to Fig. 12.6, in concentration interval (200–400 ppm), 37.5 t and 25 t of water are available for reuse in the (0.5–1.0 h) and (1.0–1.5 h) time intervals, respectively. This semi-batch behaviour is further observed in the water network shown in Fig. 12.7. Some of the water from process 3 is reused in process 1, even though process 1 is 0.5 h from completion. This would not be possible for truly batch operations, since the reuse potential of water could only be realized after the completion, and not during the course of the process.

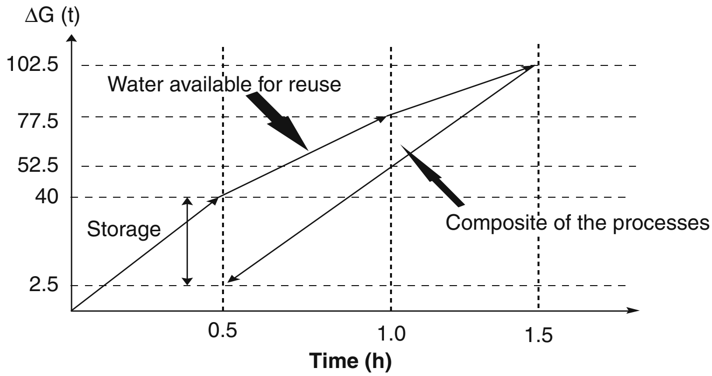


Fig. 12.6 Targeting in concentration interval (200 → 400 ppm)

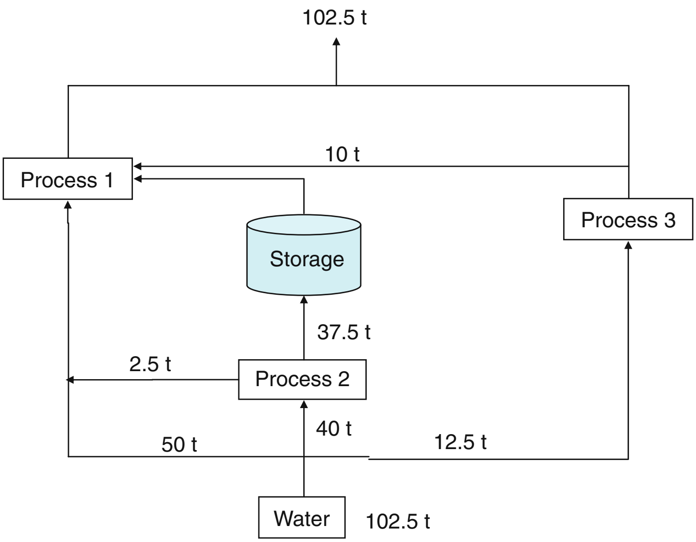


Fig. 12.7 Resultant network for the Wang and Smith problem

Secondly, the network layout showed in Fig. 12.7 shows that 12.5 t of water should be supplied to Process 3, instead of 25 t stipulated in the problem specification. This can only be true if this process does not have flowrate constraints, but has a fixed mass load. The assumption of fixed mass loads was never mentioned in the analysis. This variation of flowrate is contrary to the assumption made in targeting. During targeting it was implicitly assumed that the flowrates were fixed as shown by the calculation of water demand in each of the time subintervals.

Thirdly, the inlet and outlet concentrations were specified such that one was fixed directly and the other determined by mass balance using flowrate and mass load. However, a number of variations are possible in the way that the process constraints on quantity (or flowrate) present themselves. For instance, it could happen that there is no direct specification of the water quantity (or flow) in a particular stream, as long as the contaminant load and the outlet concentration are observed. Furthermore, the vessel probably has minimum and maximum levels for effective operation. In that case the water quantity falls away as an equality constraints, to become an *inequality constraints*, thereby changing the nature of the optimization problem.

Fourthly, the methodology suggests the need for storage, i.e. 37.5 t, when the processing times suggest otherwise. Processes 1 and 3 commence at 0.5 h from the beginning of the time horizon, which is actually the completion time for process 2. This provides a direct reuse opportunity with no requirement for storage.

The following sections of this chapter demonstrate the applicability of the above-mentioned adaptations to completely batch operations using an hypothetical example of completely batch processes drawn from experience at an agrochemical facility. Initially, time is treated as the primary constraints and concentration as the secondary constraints. Subsequently, the priority of constraints is reversed so as to demonstrate the effect of the targeting procedure on the final design. Attention is also drawn to the fact that first sequence and cyclic-state targeting are essential in completely batch operations. The cyclic-state is assumed over an extended time horizon when many batches have to be produced in multi-stage operations. The first sequence corresponds to a single batch over a relatively shorter the time horizon of interest. In order to apply the presented method to a cyclic-state behavior, the start and finish times would have to be provided for a much longer time horizon.

12.2 Graphical Analysis for Completely Batch Processes

12.2.1 Problem Statement

The problem addressed in this chapter can be stated as follows. For each water using operation, given:

(i) the contaminant mass load,
(ii) the *fixed* water requirement,
(iii) the starting and finishing times to achieve the desired effect, e.g. mass transfer, degree of cleanliness of the vessel, etc. and
(iv) maximum inlet and outlet concentrations,

determine the minimum amount of freshwater that can be achieved through the exploitation of reuse and recycle opportunities, as well as the concomitant water network. It is worthy of note that freshwater minimization is concomitant with reduction in wastewater generation. It is also assumed that the considered processes are compatible, implying that product integrity is not compromised. This implies that the issue of product mixing is excluded.

12.2.2 Time Taken as a Primary Constraints

Taking time as primary constraints implies that at each stage during the course of the analysis, the concentration constraints is readily obeyed. To illustrate the application of this analysis to completely batch processes, an hypothetical example involving liquid-liquid extraction (product washing) with water as the aqueous phase in the production of three agrochemicals A, B and C, was considered (Majozi et al., 2006). These agrochemicals were produced in batch reactors. All three reactions formed sodium chloride (NaCl) as a byproduct which was later removed from the final product. The removal of this byproduct was effected by the use of fresh water. It is worth mentioning that, although the aim of the washes was to remove NaCl, there were always traces of organics in water. In formulating the problem, however, it was assumed that the concentration of these organics was virtually negligible.

In the case of A, the reaction took place in an organic solvent which was highly immiscible with water, so that water was required solely for washing the salt. In the case of B and C, however, water was used as the reaction solvent, and a further quantity was used for washing the product. While investigating this secondary washing of B and C, it was found that the salt load removed from the product was essentially zero due to the fact that most of it had been removed with the reaction solvent water. However, it was considered that the washing step should not be discarded, as it constituted a quality control precaution in case of unforeseen process problems. The timing of the reaction and washing sequences was considered to be fixed by product requirements, which implies that there was no freedom to change the sequence to optimize the use of water.

Problem Specification for Graphical Analysis

The vessels operate in completely batch mode, implying that mass rather than flowrate of water is the relevant parameter.

Setting the concentration limits presented some conceptual difficulties. The situation for product A was straightforward, as laboratory tests had shown that acceptable product quality could only be achieved if fresh water was used at the start of the wash. The reaction solvent water for B and C was also easily specified, since the only requirement was that the salt must not precipitate, which set the salt concentration at just less than 35% by mass at the end of the reaction. Precipitation of salt led to the erosion of glass lining in the reactors. The quantity of the wash water for B and C was based on experience. It was found that using water below 400 kg led to lower product yields. This could be due to inefficient mixing in the vessel. The quantity of water was then set at 400 kg. The input concentration (which was clearly the same as the output concentration) was set as low as possible without increasing the overall water use. These have been replaced by the variable x in Table 12.2. Due to the fact that the lowest concentration for contaminated water in this example was 0.1 kg salt/kg water, x was eventually set to this value as shown in Fig. 12.8. This constituted a radical departure from the type of specification envisaged by Wang and Smith (1995).

Table 12.2 Problem specification for pinch analysis

Process	Time h	$C_{in,max}$ kg salt/kg water	$C_{out,max}$	Water kg	Salt load kg
A wash	0–3	0	0.1	1000	100
B reaction	0–4	0.25	0.51	280	72.8
B wash	4–5.5	x	x	400	0
C reaction	2–6	0.25	0.51	280	72.8
C wash	6–7.5	x	x	400	0
Total				2360	245.6

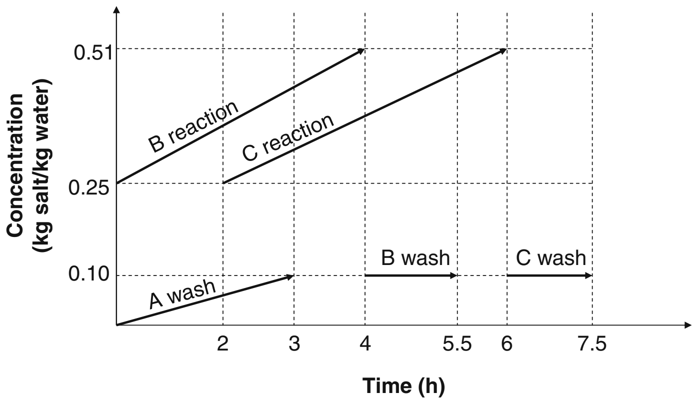


Fig. 12.8 Graphical representation of the problem specification, with x set to 0.1 kg salt/kg water (Majozi et al., 2006)

Targeting (First Sequence)

In Wang and Smith's method, the targeting procedure consists of dividing the problem into concentration intervals and time subintervals, with boundaries set by the end-points of individual processes, and grouping together the streams that are required or available for reuse in each time subinterval. In each concentration interval, water which is available is reused, where possible, in its own time subinterval. Any surplus can be reused in subsequent time subintervals in the same concentration interval, or stored for reuse in later concentration intervals. Water cannot be reused in lower concentration intervals or in previous times. Any shortfall in any time subinterval can be made up from previous time subintervals, either from the same or lower concentration intervals. Fresh water is then used once all these opportunities have been exhausted. The eventual surplus becomes the system effluent, and the accumulated fresh water make up constitutes the system intake. According to this procedure, the capacity of storage that must be provided to achieve the minimum target can be identified. It will be realized during the analysis that this storage can be traded off against water use and effluent production.

It is worthy of note that the problem solution is such that in each time subinterval, the concentration constraints is met. This implies that, as long as water is available at the right time, it can safely be reused in any of the time subintervals within the concentration interval, i.e. the secondary constraints (concentration) is met in every step of the analysis, and the primary constraints (time) guides the formulation of the final solution (target design network).

To accommodate completely batch operations, the procedure is easily modified by recognizing that water is required or available in discrete amounts, only at the beginning and end of both concentration and time intervals. The unknown concentrations (x) in the problem specification can be handled by repeating the calculations for a series of successively lower values for x, and selecting the lowest value that does not cause the overall water demand to increase.

Figures 12.9, 12.10 and 12.11 show the targeting procedure with x set at 0.1 kg salt/kg water. Figure 12.9 shows targeting in the first concentration interval. Only process A is available in this interval, Fig. 12.8, hence 1000 kg of fresh water is required. Figure 12.10 depicts targeting in the concentration boundary. Both the washes lie in this boundary, since no load is removed from the product, and the concentration of water remains constant. Since each of the washes has a water demand of 400 kg, the overall water demand at the boundary is 800 kg as shown Fig. 12.10.

It is evident from Fig. 12.10 that both B and C washes start after the completion of A wash. Moreover, the outlet concentration from the A wash corresponds to the required boundary concentration, i.e. 0.1 kg salt/kg water. Therefore, water from the A wash can be reused in B and C washes, with a surplus of 200 kg. However,

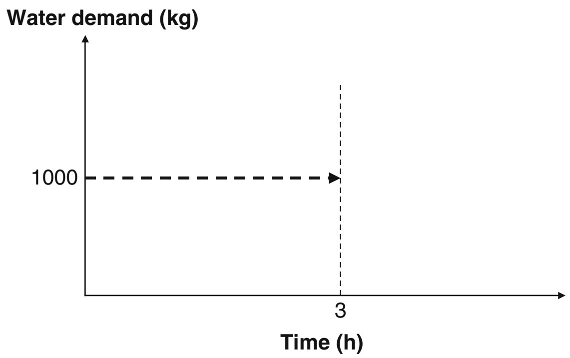


Fig. 12.9 Targeting interval (0–0.1 kg salt/kg water) (Majozi et al., 2006)

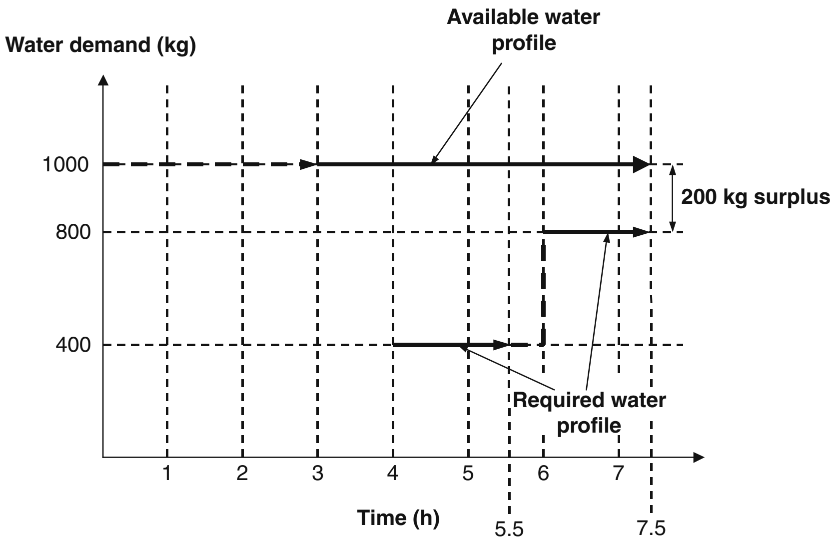


Fig. 12.10 Targeting the boundary 0.1 kg salt/kg water (Majozi et al., 2006)

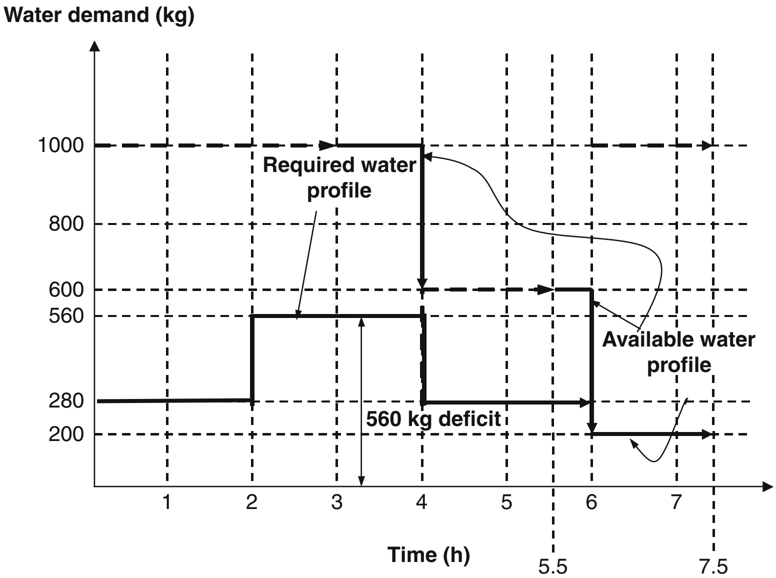


Fig. 12.11 Targeting interval (0.25–0.51 kg salt/kg water) (Majozi et al., 2006)

this water cannot be used directly since A wash finishes 1 h before the start of B wash, and C wash only starts 3 h after the A wash. The presence of the time gap between the end of the A wash and the start of the B wash signifies the need for a storage facility. If each of the washes were supplied with dedicated water, a storage capacity of 800 kg would be required. However, if water from one wash could be reused in another wash, a much smaller storage capacity of 400 kg would be required.

Figure 12.11 represents targeting in interval (0.25–0.51 kg salt/kg water). This interval, as shown in Fig. 12.8, has the B and the C reactions with an overall water demand of 560 kg. Since both these reactions start before the completion of the washing operation of product A, no reusable water is available in the reaction time subintervals. This implies that fresh water will have to be used. The accumulated fresh water demand is, therefore, 1560 kg. As this is the last concentration interval, this quantity presents itself as the target for the optimal design. This is equivalent to a 34% reduction in freshwater demand compared to the base case.

It is worth noting that interval (0.1–0.25 kg salt/kg water) did not need to be considered since no processes fall within it, Fig. 12.8.

Figure 12.12 shows the target network resulting from the foregoing analysis. As some of the input water remains in storage for reuse in later batches, effluent is less than freshwater quantity.

As mentioned earlier, there is also a possibility of reusing water from the B wash in the C wash, since their times of operation do not overlap. However, this consideration would have led to a rather different network than that shown in Fig. 12.12. The target value would still be the same.

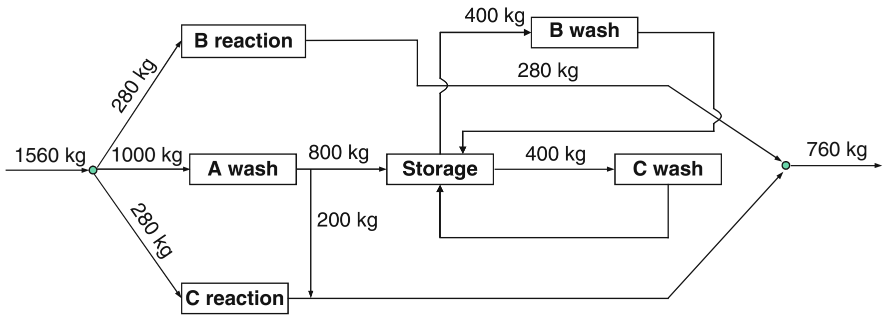


Fig. 12.12 Water reuse network resulting from pinch analysis (first sequence) (Majozi et al., 2006)

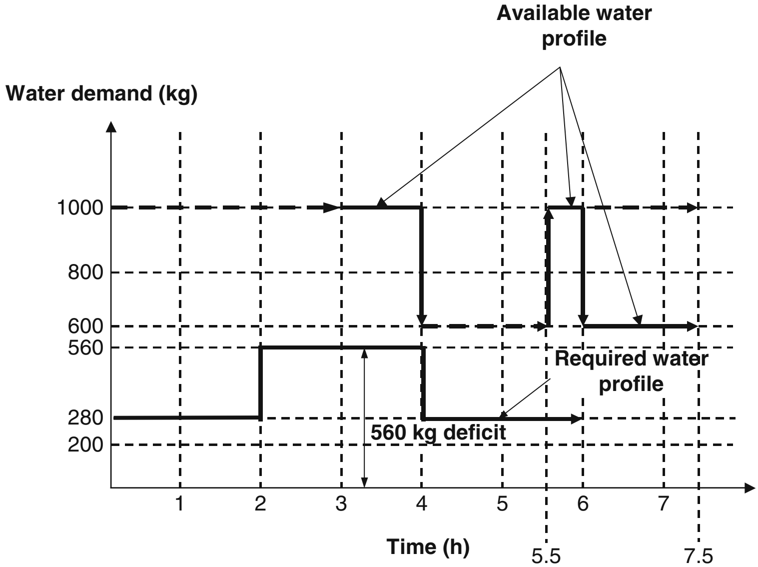


Fig. 12.13 Targeting interval (0.25–0.51 kg salt/kg water) (B wash water reuse) (Majozi et al., 2006)

Figure 12.13 represents targeting interval (0.25–0.51 kg salt/kg water) with water from the B wash reused in the C wash. Note that there is no longer any available water in the (4–5.5 h) subinterval, since it was transferred to the (6–7.5 h) subinterval, where it was used in the C wash. The amount of water remaining from the A wash is now 600 kg, since only 400 kg was reused for the washes. Nonetheless, there is still a need for 560 kg of fresh water due to the time constraints as mentioned previously.

Figure 12.14 shows the reuse network that would result from reusing water from the B wash in the C wash. The time gap between the end of B wash and the start of C wash implies that water from B wash cannot be directly reused in the C wash

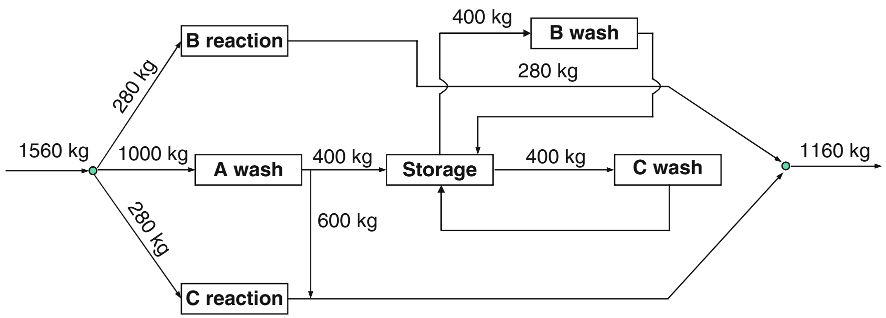


Fig. 12.14 Resultant reuse network with B wash water reused in C wash (first sequence) (Majozi et al., 2006)

(except if this vessel is readily available for water storage), hence the need for a storage facility of 400 kg capacity. It should be realized that the reduction in storage size is concomitant with an increase in effluent production as stated in Section 12.2.2 of this chapter.

Targeting (Cyclic-State Operation)

At cyclic-state the sequencing of batches might be different from the first sequence. This is usual in processes with a number of steps taking place in different vessels.

Since the resultant reuse network strongly depends on the sequence of batches during problem formulation, it is clear that a cyclic-state sequence which is different from the first sequence will culminate in a different reuse network. This entails redoing the entire problem from the beginning, and following the already presented targeting procedure. However, for a rather simple problem like the given hypothetical example, the cyclic-state reuse network can evolve from the first sequence network by exploiting obvious reuse opportunities. In this example, water from the B and the C washes can safely be reused in their corresponding reactions for the subsequent batches by either using the dedicated storage or the reaction vessels if they are readily available. This is due to the fact that the secondary constraints (concentration) is met for both the reactions. It is, therefore, evident that the primary constraints (time) can be circumvented by using storage facilities to allow the reuse of water from the previous sequence. Figure 12.15 shows the reuse network for a cyclic-state operation. It is worthy of note that the target water demand has decreased from 1560 to 1000 kg. This amounts to more than 57% reduction in freshwater demand. Whenever the reuse potential of water in a system of water using operations is increased, the inherent water demand will be reduced. The cyclic-state fresh water feed stream of 1000 kg necessitates the inclusion of the purge stream in the storage vessel to prevent the potential overflow and contaminant buildup.

Close scrutiny into the resultant reuse network as shown in Fig. 12.14 might warrant the removal of the dedicated storage facility, since, for completely batch

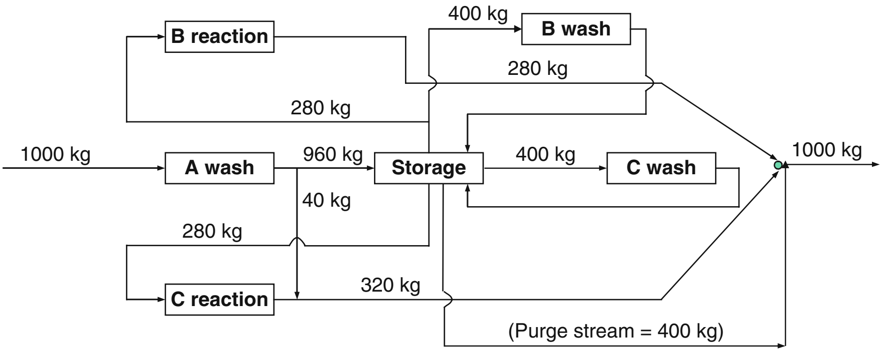


Fig. 12.15 Water reuse network (cyclic-state) (Majozi et al., 2006)

processes, the vessels can act as inherent storage facilities, as long as the timing of operations allows. This is demonstrated in the next section.

Assessment of Storage Potential for Process Vessels

The process vessels could provide part or all the necessary storage during their idle time, providing that water of the correct concentration and quantity for their next operation was available at a suitable time. To implement this idea, an *inherent storage availability diagram* (ISAD) for the set of process vessels can be constructed as shown in Fig. 12.16. The values appearing in brackets are the allowed maximum inlet and outlet concentrations in the processes. For the washes, the value of x has been set at 0.1 kg salt/kg water as mentioned previously. This means that water from the A wash meets the concentration constraints for both the B and C washes.

It can be seen from Fig. 12.16 that the B and C wash vessels can safely be used to store water from the A wash, as these washes start well after the A wash has been completed. Rather than letting the wash vessels stand idle and investing capital in a

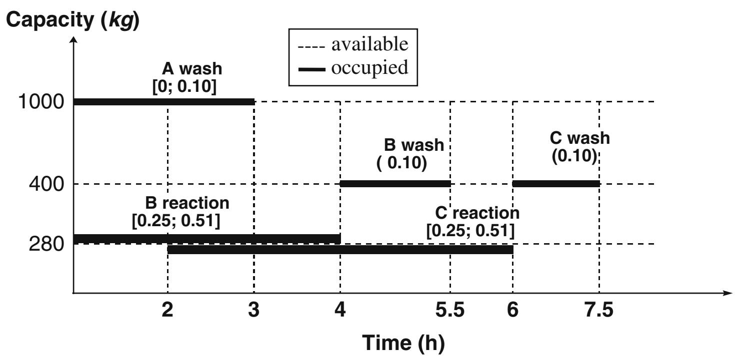


Fig. 12.16 Inherent storage availability diagram (Majozi et al., 2006)

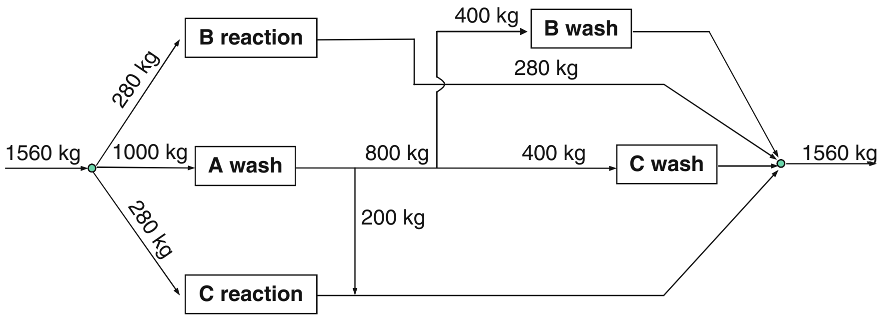


Fig. 12.17 Resultant water reuse network without storage facility (first sequence) (Majozi et al., 2006)

special storage tank, it is much more economic to remove it during the design stage and invest more in the wash vessels. Figure 12.17 shows the resultant water reuse network without any dedicated storage facility for the first sequence.

From the concentration constraints standpoint, water from the B and C washes can be recycled to the corresponding reactions. However, analysis of Fig. 12.17 shows that the washes for products B and C will be finished after the completion of the corresponding reactions for the first sequence. Therefore, the time constraints cannot be met. Nevertheless, this can be circumvented by the installation of a dedicated storage tank as shown in Fig. 12.18.

It should be realized that by increasing the reuse potential of water in this system of operations, the target water demand has decreased from 1560 to 1000 kg. It also transpired during the analysis that the value of x could be reduced to zero without increasing the overall targets for the subsequent cycles, since B and C washes do not pick up any salt, and the water could therefore be used in the A wash of the

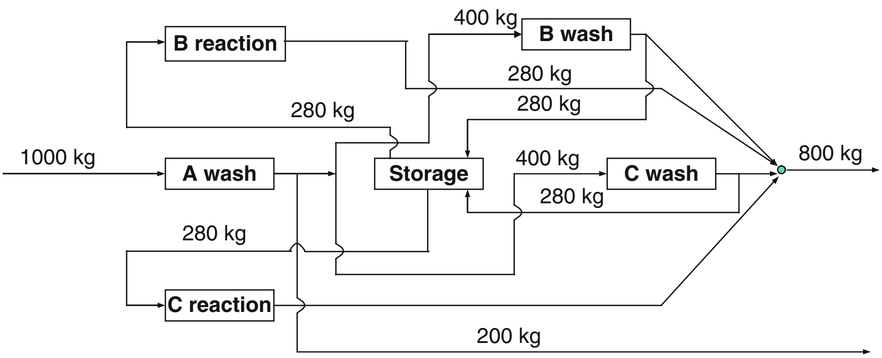


Fig. 12.18 Resultant water reuse network with process vessels used for storage (cyclic-state) (Majozi et al., 2006)

following cycle. However, this cannot be realized in the first production sequence, since the A wash precedes the others.

The foregoing analysis was mainly based on time taken as the primary constraints and concentration taken as the secondary constraints. It has been shown that this primary constraints can be bypassed by merely using storage facilities, as long as the secondary constraints is met. However, it is also possible to take concentration as the primary constraints and time as the secondary constraints. The procedure and the associated outcome of such an analysis form the subject of the following section.

12.2.3 Concentration Taken as a Primary Constraints

Figure 12.19 shows the implication of concentration taken as a primary constraints and time taken as a secondary constraints. The concentration scale represents the maximum inlet and outlet concentrations in the given processes. This concentration is increasing in the direction of the arrow on the scale. Processes A and D precede processes B and C, respectively. This implies that the secondary constraints (time) is satisfied in both case (i) and case (ii). Moreover, water from process A has concentration less than the maximum concentration allowed in process B. Therefore, in case (i) both the concentration and time constraints are met and water from process A can safely be reused in process B. However, water from process D has concentration higher than the maximum concentration allowed in process C, which implies that the primary constraints (concentration) is not met. Therefore, water from process D cannot be reused in process C, although the secondary constraints (time) is met.

It has been mentioned earlier that the time constraints can be bypassed by using the storage facilities. The concentration constraints, however, can never be readily circumvented, unless water is diluted with less contaminated or fresh water. In advanced processes, it can also be circumvented by the application of separation technologies (such as reverse osmosis).

The same hypothetical example was used to illustrate the targeting procedure for a case where concentration is treated as a primary constraints. The problem specification still remains the same as that used Section 12.2.2 (Table 12.2).

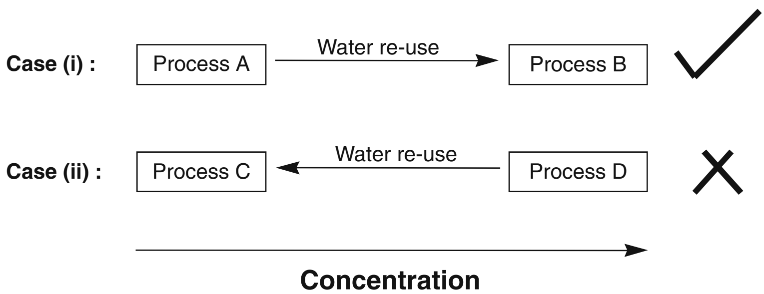


Fig. 12.19 Implication of concentration taken as a primary constraints (Majozi et al., 2006)

Targeting (First Sequence)

Instead of splitting the problem into concentration intervals and time subintervals, the problem is split into time intervals and concentration subintervals, with water demand plotted on the horizontal axis. The boundaries for time intervals and concentration subintervals are set by the process end-points. However, unlike in a case where time is taken as a primary constraints, the streams that are required or available for reuse in each concentration subinterval are plotted separately. This approach has proven to ease the analysis as will be shown later in this section.

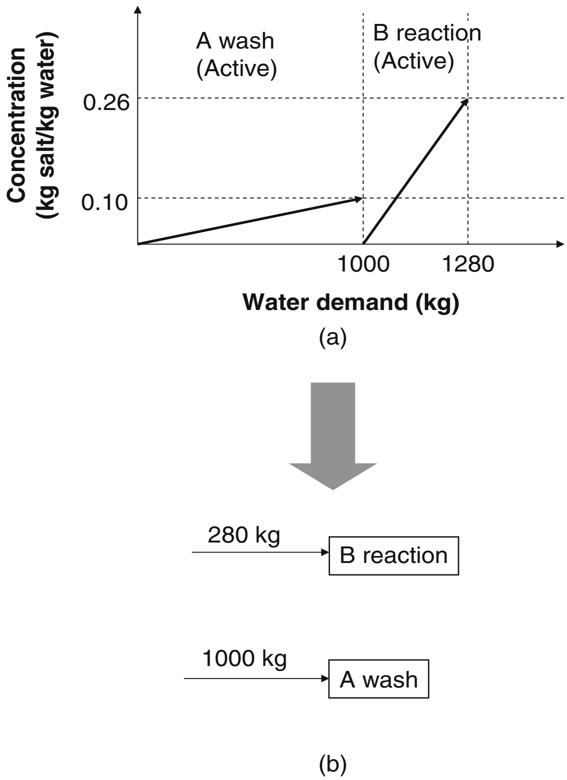
Available water from one concentration subinterval in a specific time interval is cascaded throughout the subsequent concentration subintervals, either within the same or subsequent time intervals without degeneration. This simply means that water which is available in any concentration subinterval can never be reused within the same concentration subinterval, irrespective of the time interval. If possible any surplus is transferred to higher concentration subintervals in the same time interval for reuse, or stored for reuse in later time intervals. However, this water cannot be reused in lower concentration subintervals or previous time intervals. Any shortfall within any concentration subinterval can be made up from lower concentration subintervals from previous time intervals, or from fresh water. As in the previous case, the eventual surplus becomes the system effluent and the accumulated fresh water make up constitutes the system intake.

From the analysis, the capacity of storage required to achieve the minimum target is identified. As discussed in the previous analysis, provision for storage can be traded off against water use and effluent production. The great feature of this methodology is that the network can be built during the course of the analysis without any difficulty. All the processes which have not reached completion are labeled as active processes on the concentration-water demand diagram. These active processes are represented with inlet streams only in the accompanying block diagram for each time interval. Figures 12.20a and b show targeting in the first time interval. As shown in Fig. 12.8, this time interval has two processes, viz. B reaction and the A wash, requiring 280 and 1000 kg, respectively. Since these processes are in the first time interval, fresh water will have to be used to satisfy the required water demand. The overall fresh water demand for this interval is, therefore, 1280 kg.

Figures 12.21a and b represent targeting in the (2–3 h) time interval. This interval constitutes three active processes as shown in the graph and the accompanying block diagram for this time interval. These processes are the B and C reactions, as well as the A wash. The overall water demand for this interval is thus 1560 kg, but 1280 kg is already available from the previous time interval. Therefore, only 280 kg of fresh water is required.

Figure 12.22a depicts targeting in the time interval (3–4 h). Figure 12.8 shows that only B and C reactions are active in this time interval. The A wash was completed in the previous time interval and its used water taken to storage, since there was no opportunity for its direct reuse. Figure 12.22b shows the block diagram associated with this interval. It should be noted, though, that no fresh water is needed, since the active processes started from the previous time interval.

Fig. 12.20 Targeting time interval (0–2 h) (Majozi et al., 2006)

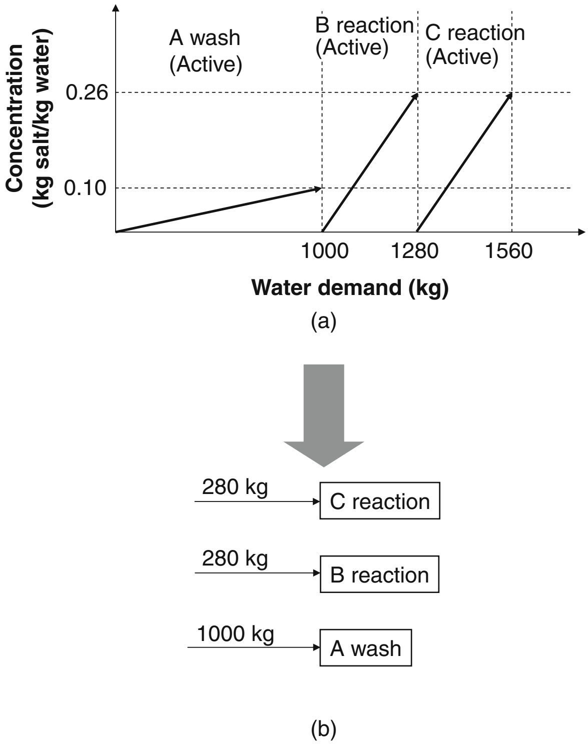


Targeting in time interval (4–5.5 h) is represented by Fig. 12.23a. Only the C reaction and the B wash are active in this time interval. The B reaction was completed in the previous time interval and its used water dispensed with as effluent due to its level of contamination. This water left this process at a concentration of 0.26 kg salt/kg water, as shown in Fig. 12.23a. As it can be inferred from Table 12.2, this concentration is above the maximum inlet concentration stipulated in any of the processes in this hypothetical example, implying that this water could never be reused, unless if it was diluted to a lower concentration.

As illustrated in Fig. 12.23a, water from the A wash is at a concentration of 0.10 kg salt/kg water, hence can safely be reused in B wash. Figure 12.23b represents the block diagram associated with this interval. It should still be realized that the active processes are characterized by not having the outlet streams.

Targeting in time interval (5.5–6 h) is shown in Fig. 12.24a. The associated block diagram is depicted in Fig. 12.24b. Since no contaminant load is removed from the B wash, dispensing with used water as effluent would certainly amount to inefficient use of available mass transfer driving forces in the system, as this water could still be reused in the next batch cycles if not reusable in the subsequent time intervals within

Fig. 12.21 Targeting time interval (2–3 h) (Majozi et al., 2006)



the current cycle. Therefore, this water was recycled back to the storage facility in order to avail it for any possible reuse in the next time interval or subsequent batch cycles.

Targeting in the last time interval is illustrated in Figs. 12.25a, b. According to Fig. 12.25a, water from the B wash can be reused in the C wash, however, the existence of a 0.5 h time gap between these processes, as shown in Fig. 12.8, warrants the use of the storage facility. This is clearly shown in Fig. 12.25b which is the associated block diagram for this time interval. Water from the C reaction also had to be disposed of as effluent due to its high contaminant load. The resultant profile for this interval is demonstrated in Fig. 12.26a, and the final water reuse network is given in Fig. 12.26b. Since targeting was aimed at the first batch sequence, water from the B and the C wash was also disposed of as effluent.

The fresh water target for the first sequence of this system of batch processes is, therefore, 1560 kg. This target is exactly the same as that obtained in Section 12.2.2 when time was taken as the primary constraints.

Targeting (Cyclic-State)

As mentioned early on in this chapter, the cyclic-state sequence is generally different from the first sequence. This means that the entire targeting procedure to get the

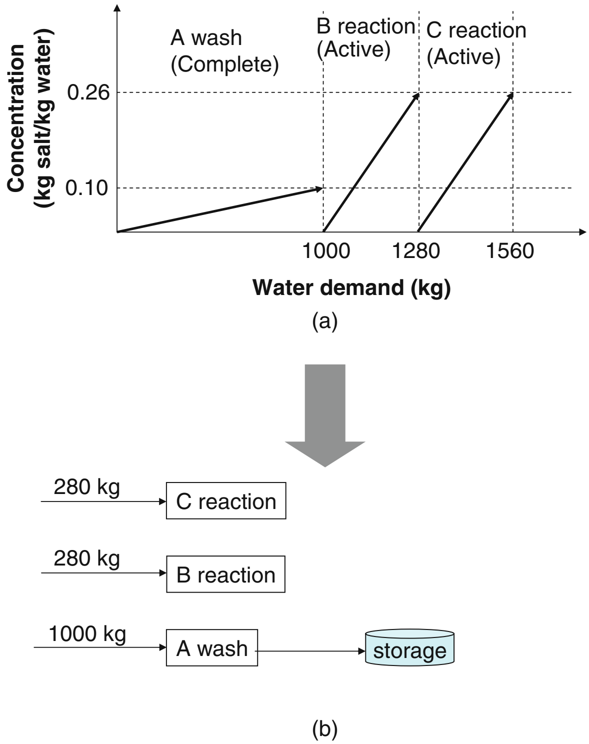


Fig. 12.22 Targeting time interval (3–4 h) (Majozi et al., 2006)

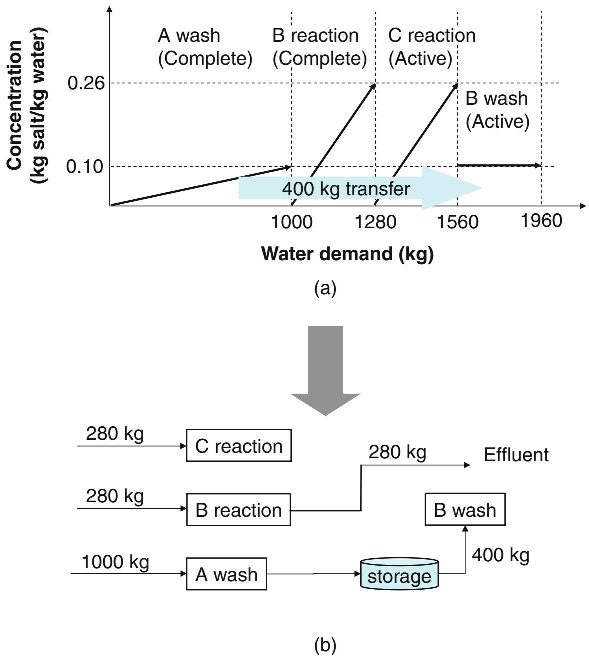


Fig. 12.23 Targeting time interval (4–5.5 h) (Majozi et al., 2006)

Fig. 12.24 Targeting time interval (5.5–6 h) (Majozi et al., 2006)

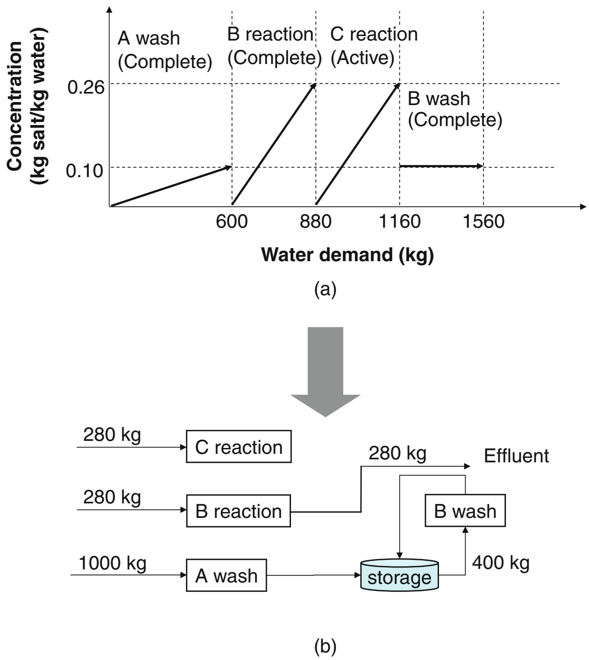
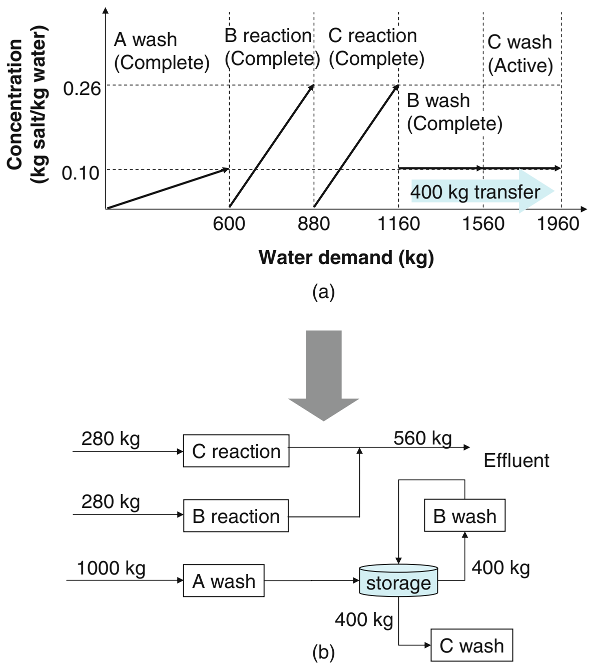


Fig. 12.25 Targeting time interval (6–7.5 h) (Majozi et al., 2006)



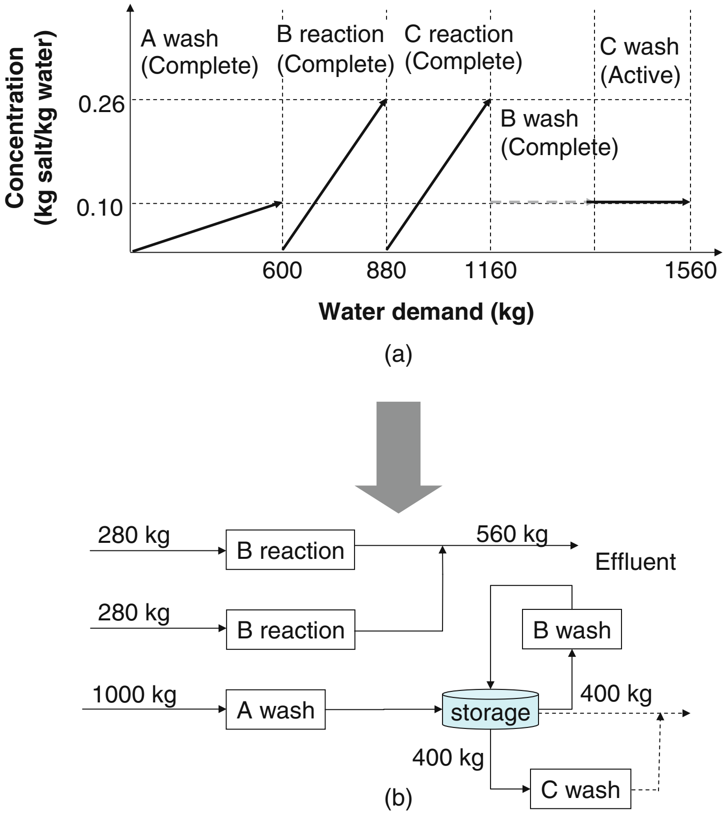


Fig. 12.26 Resultant profile for time interval (6–7.5 h) (Majozi et al., 2006)

target for a cyclic-state operation must be repeated. However, for a simple problem, like the one given in this hypothetical example, the target for a cyclic-state operation can be set by exploiting the obvious reuse and recycle opportunities. The same results as those obtained in Section 12.2.2 are also obtained when concentration is treated as a primary constraints. The use of the ISAD also yields the same results as those obtained in Section 12.2.2 for cyclic-state operation using process vessels for inherent storage.

12.3 A Brief Comparison Between Graphical and Mathematical Approaches

As aforementioned the main drawback of graphical techniques is their inherent necessary condition that the schedule should be known beforehand so as to reduce the dimensionality of the problem. This condition allows the analysis to be conducted in 2 dimensions as traditionally encountered in graphical techniques. As a result, there is currently no graphical method that treats time as a variable. Mathematical methods, on the other hand, are readily equipped with treating time as a variable, since they are dimensionally unconstrained. Unfortunately, fixing time a priori is likely to yield suboptimal results due to reduced degrees of freedom in optimization. If tasks are allowed to change their position in time, i.e. the schedule is flexible, different wastewater reuse opportunities might become possible. To illustrate this assertion, the following example originally given by Wang and Smith (1995) is considered.

The example consists of three water-using operations with maximum inlet and outlet concentrations given in Table 12.1. Over the time horizon of 90 min each operation must operate once and minimum wastewater generation must be found. In its original form, the problem assumes a fixed schedule. In the illustration, the target is firstly identified for a predefined schedule. Then the problem is solved again with the schedule allowed to change in accordance with achieving the minimum water target. In the first instance the graphical technique for truly batch operations as presented in this chapter is used. In the second instance the mathematical formulation presented in Chapter 4 is used to find the wastewater target. The problem description for the second instance is shown in Table 12.3, where only duration of operations is given instead of prescribed starting and ending times.

The resulting water reuse for the first instance is given in Fig. 12.27. The wastewater target for the three operations over the time horizon is 107.5 t. The reader is reminded at this point that the target of 102.5 t reported by Wang and Smith (1995) is based on semi-continuous, and not a truly batch, behaviour. In a truly batch operation, the target of 107.5 t is the global minimum. In Fig. 12.27 operation 1 reuses water from operation 2, once operation 2 has finished operating at 30 min. As can be seen operation 1 does not recycle or reuse any of its water. This is because the ending time of the operation is at the end of the time horizon. It is important to note that the reuse of water from operation 2 to operation 1 obeys not only the maximum inlet concentration constraints but also the inherent time constraints of a batch plant. Similarly, water from operation 3 has no opportunity for reuse within the time horizon, since it does not coincide with the start of another operation.

The resulting water reuse and operation schedule for the second instance, i.e. flexible schedule, is shown in Fig. 12.28. In this instance the wastewater target has been decreased from 107.5 t, obtained in the previous instance, to 102.5 t. Operation

Table 12.3 Data for illustrative example (2nd instance)

Unit	Max water (t)	Max outlet concen. (ppm)	Max inlet concen. (ppm)	Duration (min)	Contam. mass load (kg)
1	100	400	100	60	30
2	80	200	0	30	16
3	50	200	100	30	5

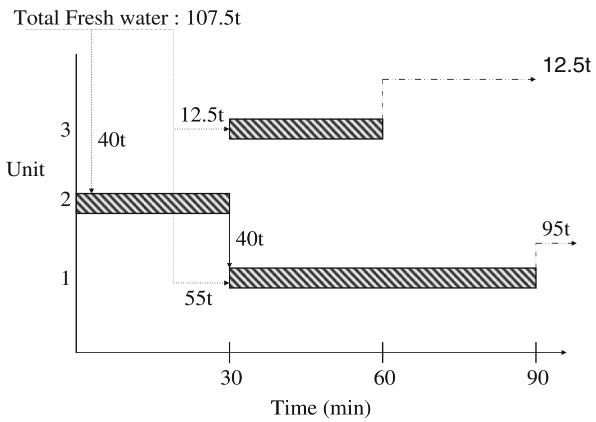
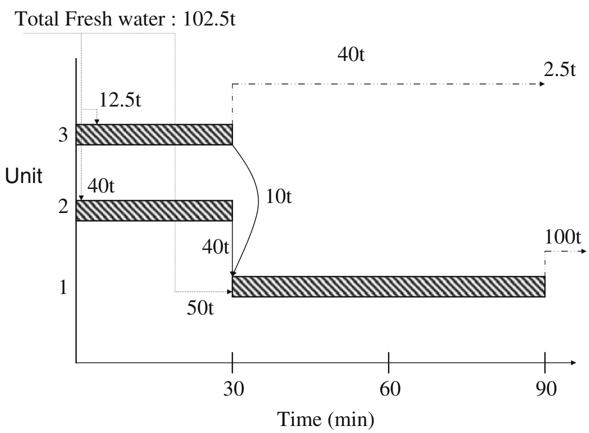


Fig. 12.27 Schedule for first instance

Fig. 12.28 Schedule for second instance



3 has moved to an earlier time within the time horizon and now starts at the same time as operation 2. Consequently, operation 1 now receives wastewater from both operations 2 and 3.

This simple example illustrates that the wastewater target for a plant is dependent on the schedule of the tasks within the time horizon. It also illustrates that the least amount of wastewater generation can only be achieved if the schedule is flexible. This implies that time is treated as a variable instead of a parameter. Important to note is that the production over the time horizon has not been negatively affected by the changing of the schedule, both wastewater generation targets and production targets are achieved.

12.4 Concluding Remarks

A new graphical technique for water minimization in batch processes has been presented. The presented technique takes into account that batch processes are constrained in both time and concentration. Applying the technique to an agrochemical facility comprising of three processes culminated in more than 30% water savings for the single batch operation and more than 50% water savings for the cyclic-state operation. The use of processing units as potential storage vessels has also been demonstrated using a so called *inherent storage availability diagram* (ISAD). In applying this technique, the design engineer can choose either concentration or time as a primary constraints. Reversing the priority of time and concentration constraints has proven not to have any effect on the water target, hence the choice of which methodology to apply in a given situation is at the discretion of the design engineer. However, the fact that time taken as a primary constraints splits the problem into concentration intervals and time subintervals, makes it a better choice for a problem with a smaller number of concentration intervals. Similarly, concentration taken as a primary constraints is a better choice for a problem with a smaller number of time intervals. The technique presented in this chapter is applicable to batch processes with single contaminant streams, which could be cited as a common drawback of graphical methods. The last section of the chapter demonstrates the limitation of graphical techniques on the quality of the solution through the use a simple literature example.

12.5 Exercise

Task: Use the graphical analysis presented in this chapter to the Wang and Smith (1995) problem and verify the result shown in Fig. 12.27.

References

El-Halwagi, M.M., Manousiouthakis, V., 1990. Simultaneous synthesis of mass-exchange and regeneration networks, AIChE J., 36(8): 1209–1219

Foo, C.Y., Manan, Z.A., Yunus, R.M., Aziz, R.A., 2004. Synthesis of mass exchange network for batch processes – Part I: Utility targeting. Chem. Eng. Sci., 59(5): 1009–1026

Gouws, J.F., Majozi, T., 2009. Usage of inherent storage for minimisation of wastewater in multipurpose batch plants. Chem. Eng. Sci., 64: 3545–3554

Gouws, J.F., Majozi, T., Gadalla, M., 2008. Flexible mass transfer model for water minimization in batch plants. Chem. Eng. Process., 47: 2323–2335

Grau, R., Graells, M., Corominas, J., Espuna, A., Puigjaner, L., 1996. Global strategy for energy and waste analysis in scheduling and planning of multiproduct batch chemical processes. Comp. Chem. Eng., 20(6/7): 853–868

Hallale, N., 2002. A new graphical targeting method for water minimization. Adv. Environ. Res., 6: 377–390

Jödicke, G., Fischer, U., Hungerbühler, K., 2001. Wastewater reuse: a new approach to screen for designs with minimal total costs. Comp. Chem. Eng., 25: 203–215

Kemp, I.C., Deakin, A.W., 1989. The cascade analysis for energy and process integration of batch processes, Part 1, Chem. Eng. Res. Des., 67: 495–509

Kiperstok, A., Sharratt, P.N., 1995. On the optimization of mass exchange networks for the removal of pollutants. Trans. IChemE, 73b: 271–277

Linnhoff, B., Flower, J.R., 1978. Synthesis of heat exchanger networks, AIChEJ., 24(4): 633–641

Majozi, T., Brouckaert, C.J.B., Buckley, C.A.B., 2006. A graphical technique for wastewater minimisation in batch processes. J. Environ. Manag., 78(4): 317–329

Obeng, E.D.A., Ashton, G.J., 1988. On pinch technology based procedures for the design of batch processes. Chem. Eng. Res. Des., 66: 255–259

Olesen, S.G., Polley, G.T., 1997. A simple methodology for the design of water networks handling single contaminants. Trans. IChemE, 75a: 420–426

Sanmartí, E., Friedler, F., Puigjaner, L., 1998. Combinatorial technique for short term scheduling of multipurpose batch plants based on schedule-graph representation. Comp. Chem. Eng., 22(Suppl.): S847–S850

Takama, N., Kuriyama, T., Shiroko, K., Umeda, T., 1979. Optimal water allocation in a petroleum refinery. Comput. Chem. Eng., 4: 251–258

Vaselanak, J.A., Grossmann, I.E., Westerberg, A.W., 1986. Heat integration in batch processing. Ind. Eng. Chem. Process. Des. Dev., 25: 357–366

Wang, Y.P., Smith, R., 1994. Wastewater minimization. Chem. Eng. Sci., 49(7): 981–1002

Wang, Y.P., Smith, R., 1995. Time pinch analysis, Trans. IChemE, 73a: 905–914

Wang, Y.P., Smith, R., 1995. Waste minimization with flowrate constraints, Trans. IChemE, 73: 889–904

Yao, Z.L., Yuan, X.G., 2000. An approach to optimal design of batch processes with waste minimization. Comput. Chem. Eng., 24: 1437–1444.

Index

T. Majozi, *Batch Chemical Process Integration*, DOI 10.1007/978-90-481-2588-3,