A Mixed-Integer Linear Programming Model for Optimizing the Scheduling and Assignment of Tank Farm operations

Sebastian Terrazas-Moreno[a], Ignacio E. Grossmann[a*], John M. Wassick[b]

[a]*Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh PA 15232, USA*
[b]*The Dow Chemical Company, 1776 Building, Midland, MI 48674, USA*

Abstract

This paper presents a novel mixed-integer linear programming (MILP) formulation for the Tank Farm Operation Problem (TFOP), which involves simultaneous scheduling of continuous multi-product processing lines and the assignment of dedicated storage tanks to finished products. The objective of the problem is to minimize blocking of the finished lines by obtaining an optimal schedule and an optimal allocation of storage resources. The novelty of this work is the integration of a tank assignment problem with a scheduling problem where a dedicated storage tank has to be chosen from a tank farm given the volumes, sequencing, and timing of production of a series of products. The scheduling part of the model is based on the Multi-operation Sequencing (MOS) model by Mouret et al., (2011). The formulation is tested in three examples of different size and complexity.

Keywords: Tank farm operation, production scheduling, tank assignment, multi-operation sequencing.

Introduction

Chemical manufacturing sites ship finished products to customers using different modes of transportation (MOT) such as railcars, tank trucks, and pipelines. These MOTs are usually loaded from or connected to storage tanks. Consequently, all finished products have to be fed from the process into the storage tanks before being shipped to customers. This type of operation imposes the need for available storage space at all times in order to avoid unnecessary shut-downs of the upstream chemical process. When these shutdowns occur, they are said to be a result of storage tanks blocking the process. If the chemical process produces several products and each one requires dedicated tanks, the product-tank assignment and the processing schedule at the finishing lines determines how efficiently the storage space is used. An inefficient assignment of tanks or processing schedule can result in blocking the production of certain products, even when there is plenty of available storage space in tanks assigned to other products.

In this paper, we consider a multiproduct manufacturing facility that includes finishing lines connected to a number of storage tanks from which products are shipped to customers using different modes of transportation (MOT). The tanks are dedicated to one product, meaning that once a product is assigned to a tank, it cannot be used for another product. Cleaning of tanks to store different products is not allowed. We are concerned with finding the best possible assignment of products to dedicated storage tanks, and the best processing schedule at the finishing lines in order to minimize the unallocated production. For scheduling purposes, there is a set of production orders that requires the processing of a certain amount of product in the finishing lines. Orders have a known release date and all are due at the end of the time horizon. The release date corresponds to the moment when a certain amount of an unfinished product becomes available for processing in the finishing lines. These lines can process the order immediately and feed the storage tanks, or delay the order for a while until there is available storage space. The main decisions in this problem are the tank-product assignment and the scheduling of processing orders. The set of storage tanks included in the production facility is collectively known as a Tank Farm, and therefore, the problem outlined above is referred to in this paper as the Tank Farm Operation Problem (TFOP).

The paper by Sharda and Vazquez (2009) illustrates the relevance of the tank farm operation problem (TFOP) for the Dow Chemical company. The authors describe the development of a Decision Support System to evaluate the operation of a tank farm at a chemical production site in Freeport, TX. The system they propose is based on Discrete Event Simulation (DES). In discrete event models the system changes states as events occur and only when those events occur (ExtendSim, 2007). DES works by representing the occurrence of an event by generating and passing items among the elements of the simulation model. Cassandras et al. (1993), Banks et al. (2005), and ExtendSim (2007) are useful references on Discrete Event Systems and Discrete Event Simulation. Sharda and Vazquez (2009) evaluate the operation of a tank farm in a Dow Chemical site that consists of more than 80 tanks for storing 60 product families, processed in 6 finishing lines. Their simulation model captures complex operating constraints, such as the effect of recycle lines on the simultaneous loading and unloading of tanks, operational logic when dealing with delayed processing orders, and the stochastic nature and dynamics of the operation of loading from tanks unto the different modes of transportation (MOT). Since the approach is based on simulation, it is not intended for finding an optimal resource allocation or production schedule; it is a tool for evaluating different storage tank allocations for a given production schedule.

Other authors have used this type of DES as a support tool for operating a tank farm. Chen et al. (2002) from the Mathematical Modeling Group at BASF provide another example of a DES study of logistics in a chemical plant. The tank farm problem is also relevant in crude refining operations. For instance, Stewart and Trierwiler (2005) carried out a study of the tankage requirement in different operational scenarios for the Kuwait National Petroleum Company. Their main tool is DES which they combine with Linear Programming (LP). Chryssolouris et al. (2005) present an integrated simulation based approach that manages scheduling, tank farm, inventory, and distillation operations in a refinery. Their approach is based on generating random solutions within a given search space and evaluating them using the simulation model.

Mathematical programming is another approach to the Tank Farm Operation Problem (TFOP). Optimization methods can determine the best possible tank allocation and/or production schedule within a given search space, as opposed to simulation tools that require the allocation and schedule to be specified. The disadvantage of this approach is that it requires operational constraints to be expressed as algebraic equations. Many of the operational constraints of TFOP have to be simplified in order to express them with algebraic equations for the optimization algorithms. Even with this limitation, optimization approaches have been successfully used in problems related to storage tank allocation and tank transfer scheduling.

Hvattum et al. (2009) address the storage tank allocation problem (TAP) in a maritime bulk shipping operation. A ship is equipped with a set of storage tanks that are loaded and unloaded according to a scheduled route. This problem has similar constraints as the tank farm operation problem we address in this paper; each tank holds only one type of product and several tank-product assignments are infeasible due to structural or safety constraints. The loads received and delivered at different ports are similar to product loading and unloading to storage vessels in the tank farm according to a production and shipping schedule. A Mixed-integer Linear Programming (MILP) model is used to either test the feasibility of a shipping route, to minimize tank cleanup time, or to maximize the vacant space in storage tanks. The main difference between the TAP and the TFOP is that the TFOP deals with a continuous production system, whereas the TAP deals with a limited number of discrete loading and unloading events. The work by Ha et al. (2000) is a good example of a research topic related with storage tank allocation, namely, the optimization of intermediate buffer sizing and allocation in multi-product batch process systems. Ha et al. (2000) determine the location, number, and size of storage tanks with the objective of minimizing the makespan. As opposed to the Tank Farm Operation Problem (TFOP), storage vessels can be shared by different types of batches. Vecchietti and Montagna (1998) address a similar problem. On a similar subject, there are mathematical programming approaches that deal simultaneously with production scheduling and storage considerations. Shaik and Floudas (2007) where the authors consider the short-term scheduling of continuous processes using a state-task-network representation with unit-specific event-based continuous-time formulation that rigorously models several storage requirements such as dedicated, flexible, and no intermediate storage. Sundaramoorthy and Maravelias (2008) present a simultaneous batching and scheduling formulation for a multiproduct multistage process that includes storage detailed storage constraints related to the maximum time a batch can spend in storage, and that acknowledges that in multistage processes batches of the same product cannot share the same storage vessel, making the size and number of storage units and important consideration. None of the last two works, however, considers the assignment of products to dedicated storage tanks among a preexisting tank farm to be a degree of freedom. In this fact lies a significant part of the contribution of the present paper where the tank assignment problem is simultaneously solved with the scheduling problem.From the above review, we can conclude that Discrete Event Simulation (DES) is the approach that has received the most attention for addressing the TFOP in process industries. Mathematical optimization has been mostly aimed at related problems such as buffer tank allocation in multiproduct batch scheduling, or to assignment problems such as in the case of the tank allocation problem in maritime operations. Some authors (Zeng and Yang, 2009; Chen et al., 2002) have pointed out that the number of variables and constraints, the operational complexity, and the stochastic nature of logistic processes involved in tank farm management produce either intractable or oversimplified mathematical optimization models. Zeng and Yang (2009) propose integrating simulation and optimization for solving the TFOP. Their argument is that an approximate optimization model is sufficient for obtaining good solutions that can be evaluated using the simulation model in a second step. They use neural networks and genetic algorithms for the optimization module.

The objective of this paper is to present an optimization method based on mathematical programming, namely Mixed-Integer Programming (MILP), for solving the TFOP. Our model of the TFOP includes scheduling of orders that arrive to the finishing lines as well as the optimal tank allocation. We use the description of the tank farm operation given by Sharda and Vazquez (2009), and try to incorporate in our MILP model as much of the operational details as possible. This paper does not include simulation by DES. Nevertheless, we believe that the output of the optimization model we propose could be evaluated, validated, and communicated using DES. The novelty of our work is the management of the system of storage tanks (i.e., the tank farm) as part a decision variable in a single stage production process. From the previous literature, we can see that the most closely related problems in the scheduling literature contained in the works by Shaik and Floudas (2007) and Sundaramoorthy and Maravelias (2008) do not consider the decision making process involved in assigning a tank or set of tanks from an existing tank farm as part of the degrees of freedom in a scheduling problem. It should be clear to the reader that although we choose a suitable existing scheduling model (Mouret et al., 2011) to carry out the integration of the tank assignment and the scheduling problems, the novelty of our work is not in the scheduling formulation itself but rather in the integration of the tank assignment and scheduling problems.

In the following sections of this paper we give a more detailed description of the tank farm problem, state the optimization problem, discuss the mathematical model, and apply the formulation to three case studies. We end this paper with a summary of our findings and a discussion of possible future work.

Problem Statement

In this section we describe the tank farm problem as defined in this paper. The downstream section of a chemical production facility includes a set M of continuous finishing lines that represent the last step in the manufacturing of a set J of products, along with a set K of storage tanks (the tank farm) from where the products are shipped to the consumers. The shipping operation uses different modes of transportation, usually railcars, tank trucks, or pipelines. Figure 1 (Sharda and Vazquez, 2009) shows a representation of this system.

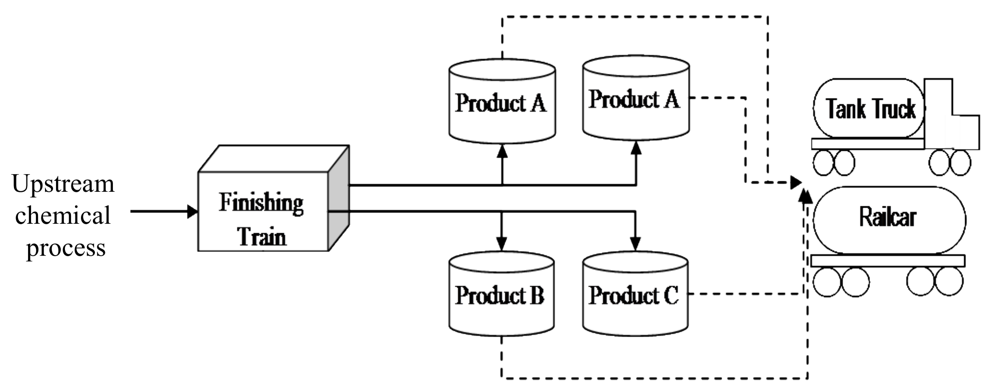

Figure 1. Downstream section of chemical production process

The processing step in lines $m \in M$ is carried out according to processing *orders*. A processing order is generated when a batch of unfinished product from the upstream chemical plant is ready to be processed by the finishing lines, which operate continuously. Each order is single product. The parameter $d_{o,j}$ specifies the amount of product j to be produced according to order o. Each order $o \in O$ has a release date rd_o

that corresponds to the moment when the unfinished product is available for processing in the finishing lines, and when the corresponding production order gets generated. All orders are due at the end of the operating horizon. If an order cannot be processed at its release date, its production can be delayed as long as it can still meet the due date. The production rate of each product $j \in J$ in line m, $rate_{j,m}$, is a known deterministic parameter. After its release date, rd_o, the order o is processed in the finishing lines and sent to storage in one of the $k \in K$ tanks of the tank farm. At the beginning of the time horizon, some or all of the tanks are empty.

It should be pointed out that for simplicity we assume that the tanks are initially empty. In practice, however, it is clear that most likely several tanks may be partially filled at the initial point, unless they have been emptied for cleaning and maintenance proposes. For modeling purposes, however, it is straightforward to handle the case when several tanks are partially filled. In particular, once a tank k has been allocated to store a product j, it remains dedicated to this product throughout the operating horizon considered in this paper. In this way, by imposing this additional constraint the discrete decision of assigning the product j to a tank k is actually removed thereby reducing the degrees of freedom in this problem.

There are a number of operational characteristics that make the management of a tank farm a complex problem. The processing of certain products might not be possible in every finishing line, and a product can only be transferred from a line to a tank if there is piping or some other type of connection between them. Simultaneous loading and unloading of tanks is not allowed. Finally, shipping can be done using different modes of transportation (MOTs), namely, railcars and tank trucks that are loaded in batches at specified times that correspond to the arrival of available MOTs.

The problem can be summarized as follows:

Given are:

- A finite time horizon

- A set of finishing lines and their operational characteristics:
    - Subset of products that can be processed in it
    - Processing rate for each product
    - Subset of tanks that the line is connected to, and to which it can transfer material
- A set of production orders with the following information:
    - A product to be processed (orders are single product)
    - Quantity of product to be processed
    - Release date
    - A due date corresponding to the end of the operating horizon
- A set of products
- A set of storage tanks and their characteristics:
    - Maximum capacity (volume or mass)
    - Subset of products that are compatible with its operating conditions
    - Frequency of unloading to MOT
    - Rate of transfer to MOT during unloading

The problem is to determine:

- The assignment of products to dedicated storage tanks
- The assignment of orders to finishing lines
- The start and duration of the processing time for each order
- The tank or tanks to which each order will be transferred to

Subject to:

- Assignment constraints:
    - One product per tank
    - Maximum and minimum number of tanks for a product
    - Physical and chemical compatibility of finished product and tank conditions
    - Constraints given by existing connections between finishing lines and storage tanks

- Mass balance constraints
    - The rate of processing of an order is equal to the rate of transfer to storage tanks
    - Accumulation in storage tanks is equal to all transfers from lines minus all quantity loaded to MOTs
- Scheduling constraints
    - No two orders can be processed simultaneously in the same line
    - An order can only be processed after its release date
- Tank operation constraints
    - Simultaneous loading and unloading is not allowed
    - Unloading operations to MOTs occur with predetermined frequency
    - Unloading operations to MOTs have a predetermined maximum duration

With the objective of:

- Minimizing unallocated production that results from blocking finishing lines by unavailability of storage space

The simplifications with respect to realistic tank farm operation are as follows:

- Unloading operations occur at the same time in all tanks. Otherwise, the unloading of each tank at the end of each day could potentially require a priority slot, and the problem size would increase considerably.
- Simultaneous loading and unloading of tanks is prohibited.
- Modes of transportation operate in semi continuous mode only; there are no continuous MOTs.
- Changeover times are neglected for the scheduling.

Continuous time scheduling in the Tank Farm Operation Problem (TFOP)

As we stated in the previous sections, production orders can be delayed. This allows for a production order to be scheduled some time after its release date, and allows optimization of the processing schedule at the continuous finishing lines in order to make the most of the available storage space given by the tank assignment decisions. An important characteristic of the finishing process is that once an order has started, it has to be processed completely or the unprocessed quantity has to be declared as unallocated product. This fact is a result of the operational logic shown in Figure 2.

There are alternative models that can be used for scheduling continuous parallel production lines. Since we are interested in the material balance in the storage tanks, we need to consider the transfer of different products from several parallel finishing lines. The simplest alternative to do so is to use a discrete time formulation. The state-task-network (Kondili et al., 1993) and the resource-task-network (Pantelides, 1994) are the most general discrete time formulations for batch processes. Since the process at hand is single-stage and continuous, a much simpler formulation where at most one production order can be assigned to each time interval and each line could be used. The transfer to storage tanks would be equal to the processing rate, and the unloading of tanks to MOTs could also be specified for some of the time intervals. The balance of the inventory tank could be easily calculated at the end of each time interval. The drawback of this simple discrete time formulation is that when a large number of time intervals are needed, the problem may become intractable.

Continuous time scheduling models are common nowadays since they can potentially decrease the combinatorial complexity that results from discrete time models (Floudas and Lin, 2004). Erdirik-Dogan and Grossmann (2008) present a model for simultaneous planning and scheduling of continuous parallel production lines that divides the operating horizon into planning time periods. The inventory mass balance is calculated at the beginning and at the end of each time period. A slot-based MILP scheduling problem in continuous time is solved within each time period. This model has a very natural way for incorporating sequence-dependent changeovers that has been extended by Lima et al. (2011) and Kopanos et al. (2011) to consider production changeovers across time periods. Yet another type of continuous time models relevant to the TFOP comes from the study of tank transfer and crude oil scheduling problems in the refining industry (Furman et al., 2007; Mouret et al., 2009; Mouret et al., 2011). Among these alternative models, we use the Multi-operation Sequencing (MOS) model described by Mouret et. al (2011). We justify our decision as follows. On one hand, we exclude a Discrete-time formulation following the arguments of Floudas and Lin (2004) that state that a discrete-time formulation would lead to very large models that could quickly become intractable, especially since the problems considered in these paper span several weeks but require small time intervals given the short duration of some operations like shipping events that last only for a few hours. On the other hand we prefer the MOS model over the one by Erdirik-Dogan and Grossmann (2008) since this last one would most likely require a larger number of time slots, given than the time horizon has to be divided into time periods and within each one a number of priority slots has to be postulated. The number of time periods would have be equal to the maximum potential number of shipping events in order to carry out the mass balances in the storage tanks, and within each one a number of slots proportional to the unprocessed orders has to be postulated. In contrast, the MOS requires a number of time slots proportional only to the number of orders since there is no need to define time periods. We acknowledge that the model size is not the only relevant variable for computational efficiency, namely, there is also the tightness of the linear relaxation, but since the focus of our work is on the integration of scheduling and tank assignments, we consider that these criteria for selection of a scheduling model are well justified although a stricter selection of scheduling model would require extensive testing of several case studies using all possible types of models.

Mathematical Model

In this section we present a mathematical model that corresponds to the problem defined above where the tank assignment decisions and the scheduling decisions are solved simultaneously. The mathematical model in this paper is based on the Multi-operation sequencing (MOS) model of Mouret et. al (2011). It has been modified to account for specific considerations of the TFOP. The MOS formulation by Mouret et. al (2011) uses the idea of *operations* assigned to priority time slots that enforce a precedence of time events. Multi-operation sequencing (MOS) receives its name from the fact that several operations can be assigned to the same slot unless they are considered as *non-overlapping*. To illustrate this concept, consider the six operations and three resources shown in Table 1 (Mouret et al., 2011). Operation v_4 consumes resources r_1 and r_2, which means it cannot be performed simultaneously with either operation v_1 or v_2. Operations that share resources are termed non-overlapping.

Table 1. Set of operations and resources used by each operation

Operation	v_1	v_2	v_3	v_4	v_5	v_6
Resource	r_1	r_2	r_3	$r_1 \wedge r_2$	$r_1 \wedge r_3$	$r_2 \wedge r_3$

Figure 2 (Mouret et al., 2011) shows another way of representing operations and the non-overlapping relationship among some of them. The graph in this figure has an arc between all non-overlapping operations. The idea of a clique from graph theory is used to group subsets of non-overlapping operations. For instance, the subset $\{v_2, v_4, v_6\}$ is a clique in the graph on Figure 2. Figure 3 (Mouret et al., 2011) shows a feasible schedule with the six operations in Table 1 that illustrates the idea of multi-operation sequencing (MOS) and non-overlapping operations for a set of 4 priority time slots. For instance, note that operations v_1 and v_6 are assigned to slot 1, while operations v_2 and v_5 are assigned to slot 2. In general, let V be the set of operations, L the set of slots, $S_{v,\ell}$ the start of operation $v \in V$ in slot $\ell \in L$, $E_{v,\ell}$ the ending time, and $Z_{v,\ell}$ a binary equal to 1 if v is assigned to ℓ. Then, the basic idea of the MOS model is summarized by the following constraints:

$$\sum_{v \in V^*} Z_{v,\ell} \le 1 \qquad \ell \in L, V^* \in clique(V) \qquad \text{(MOS1)}$$

and

$$\sum_{v \in V^*} E_{v,\ell_1} \le \sum_{v \in V^*} S_{v,\ell_2} + H(1 - \sum_{v \in V^*} Z_{v,\ell_2}) \qquad \ell_1, \ell_2, \in L, \ell_1 < \ell_2, V^* \in clique(V), \qquad \text{(MOS2)}$$

where $clique(V)$ is formed by all subset of operations $V^* \subseteq V$ such that any two operation in V^* must not overlap. For instance, all operations that use a common resource would be part of a set $V^* \in clique(V)$. Note that constraint (MOS1) states that at most one operation v from a given subset of operations V^* can be assigned to a slot ℓ, while constraint (MOS2) enforces for all operations from the subset V^*, that the end time of slot ℓ_1 takes place before the start time of slot ℓ_2 if an assignment is made.

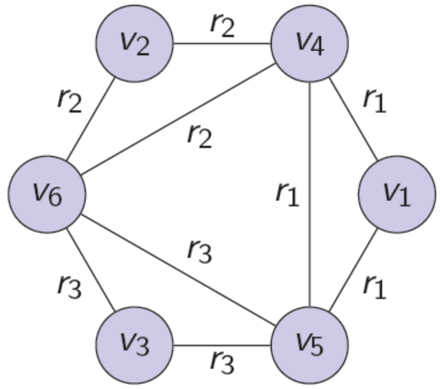

Figure 2. Non-overlapping graph

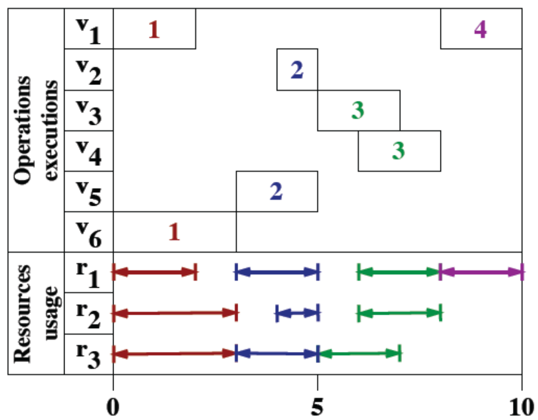

Figure 3. An illustrative solution schedule

In this paper we use the same approach by defining as non-overlapping operations those that occur in the same processing line, as well as any pair of shipping and processing operations. Shipping and processing are non-overlapping since loading and unloading of a tank cannot occur at the same time. The set V is a collection of two types of operations: processing of an order in a finishing line, and an unloading or shipping event. The first type is defined by a pair $(o,m) \in O \times M$; the second type of operation considers any shipping event $s \in S$. In set notation: $V = \{(o,m),(s) : o \in O, m \in M, s \in S\}$. The cliques that appear in equations (MOS1) and (MOS2) are $clique_m(V) = \{(o,m),(s) : o \in O, s \in S\}, \forall m \in M$. These cliques are result of defining that the processing of two orders on the same finishing line, and that loading and unloading of a tank must not overlap. As an example, consider the small system described in Table 2, where three processing orders have to be scheduled in two finishing lines. The finished products are stored in three tanks from which there will be shipped once to the customer.

Table 2. Small illustrative example

Processing orders	o_1	o_2	o_3
Product in order *i*	*A*	*B*	*C*
Finishing lines	m_1	m_2	
Storage tanks	T_1	T_2	T_3
Shipping events	s_1		

Table 3. Definition of operations in small illustrative Example

Processing operations	(o_1,m_1)	(o_2,m_1)	(o_3,m_1)	(o_1,m_2)	(o_2,m_2)	(o_3,m_2)
Shipping operations	s_1					

Figure 4 shows the non-overlapping graph that corresponds to the operations in Table 3.

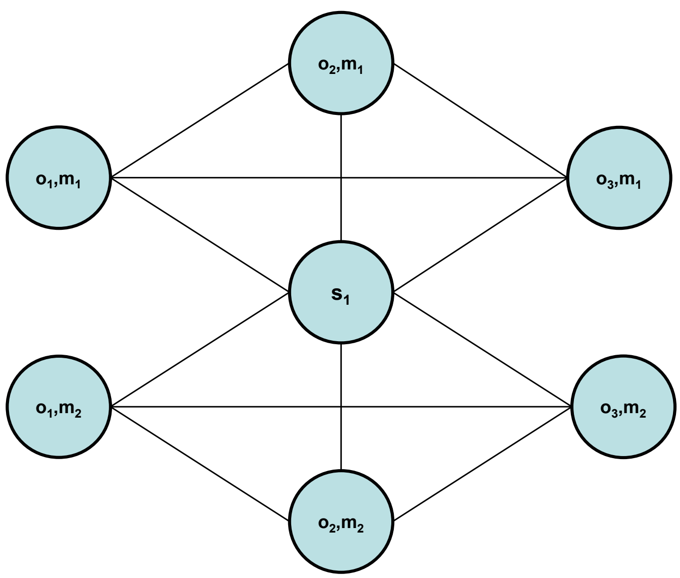

Figure 4. Non-overlapping graph in small illustrative example

There are two cliques in the graph of Figure 4:

$clique_{m_1} = \{(o_1, m_1), (o_2, m_1), (o_3, m_1), (s_1)\}$, and

$clique_{m_2} = \{(o_1, m_2), (o_2, m_2), (o_3, m_2), (s_1)\}$.

Recalling that the set of slots is L, the non-overlapping constraint (MOS1) in the small illustrative example is written below:

$$\sum_{v \in V^*} Z_{v,\ell} \leq 1 \qquad \ell \in L, V^* \in clique_m \quad \forall m \in M \, ,$$

or alternatively, if variable $Z_{v,\ell}$ is disaggregated into a variable for assigning processing operations to slots $wo_{o,m,\ell}$ and another for assigning shipping events to slots $ws_{s,\ell}$,

$$\sum_{o \in O} wo_{o,m,\ell} + \sum_{s \in S} ws_{s,\ell} \leq 1 \qquad \ell \in L, m \in M \, .$$

The latter form has the advantage that the cliques do not have to be explicitly defined, and that the constraint is described in terms of the naturally occurring sets L and M. This is the form that we use in equations (4) and (5) of the mathematical model. Finally, a feasible schedule for this illustrative case using 3 priority time slots is found in Figure 5.

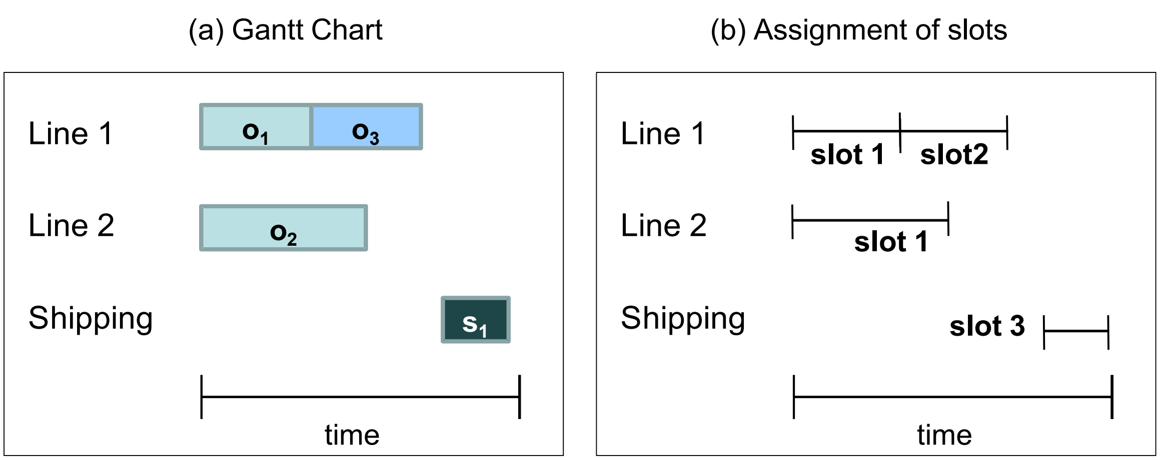

Figure 5. A feasible schedule for the illustrative example

Objective function

The objective is to minimize the unallocated product. This objective is equivalent to maximizing the amount of finished product that is allocated to the storage tanks during the time horizon. The parameter ω_o is included to give different weights to different production orders. For instance, if an order is urgent it could have a large value of ω_o. For all the results in this paper we set $\omega_o = 1, \forall o \in O$.

$$ap = \sum_{o \in O} \sum_{\ell \in L} \sum_{m \in M} \sum_{k \in K} \omega_o v^{in}_{o,\ell,m,k} \qquad \text{(obj)}$$

Scheduling constraints

Equations (1a) and (1b) establish that the start time for the processing of order o in line m at slot ℓ is zero if the order is not assigned to that line during that slot, or otherwise it must be sometime after the release date and before the end of the horizon. Equation (1c) is a constraint on the frequency of shipping events. This equation fixes the start of each event s at slot ℓ to either zero or the predetermined shipping time $stime_s$. In equation (1d) the duration of the processing operation of order o in line m can only be non-zero if has to been assigned to slot ℓ. The duration of a shipping event is either zero or less than the predetermined maximum duration of shipping events, as established by equation (1e). Equations (1f) and (1g) set the ending time of any slot to either zero if it is not assigned to processing or shipping, or to less than the operating horizon. Equations (2a) and (2b) establish that the ending time of any slot is equal to its starting time plus its duration.

$$sto_{o,\ell,m} \le wo_{o,\ell,m} H \qquad o \in O, \ell \in L, m \in M \qquad \text{(1a)}$$

$$sto_{o,\ell,m} \ge wo_{o,\ell,m} rd_o \qquad o \in O, \ell \in L, m \in M \qquad \text{(1b)}$$

$$sts_{s,\ell} = ws_{s,\ell} stime_s \qquad s \in S, \ell \in L \qquad \text{(1c)}$$

$$dro_{o,\ell,m} \le wo_{o,\ell,m} H \qquad o \in O, \ell \in L, m \in M \qquad \text{(1d)}$$

$$drs_{s,\ell} \le ws_{s,\ell} shipwin_s \qquad s \in S, \ell \in L \qquad \text{(1e)}$$

$$eo_{o,\ell,m} \le wo_{o,\ell,m} H \qquad o \in O, \ell \in L, m \in M \qquad \text{(1f)}$$

$$es_{s,\ell} \le ws_{s,\ell} H \qquad s \in S, \ell \in L \qquad \text{(1g)}$$

$$eo_{o,\ell,m} = sto_{o,\ell,m} + dro_{o,\ell,m} \qquad o \in O, \ell \in L, m \in M \qquad \text{(2a)}$$

$$es_{s,\ell} = sts_{s,\ell} + drs_{s,\ell} \qquad s \in S, \ell \in L \qquad \text{(2b)}$$

It should be noted that the sum of all processing durations $dro_{o,\ell,m}$ for each order are bounded by the sum of the product demands $do_{o,j}$ as will be specified later in constraint (6).

By constraint (3a) an operation can be assigned at most once to any slot or finishing line; by constraint (3b) each shipping event only takes place once.

$$\sum_{\ell \in L} \sum_{m \in M} wo_{o,\ell,m} \leq 1 \qquad o \in O \qquad (3a)$$

$$\sum_{\ell \in L} ws_{s,\ell} \leq 1 \qquad s \in S \qquad (3b)$$

Constraint (4) enforces that no overlapping operations are assigned to the same slot. It was described in more detail in the previous sections.

$$\sum_{o \in O} wo_{o,\ell,m} + \sum_{s \in S} ws_{s,\ell} \leq 1 \qquad \ell \in L, m \in M \qquad (4)$$

According to the idea of priority slots (Mouret et al., 2009), slot ℓ_1 in equation (5) has a higher priority than slot ℓ_2 in the sense that it takes place before or at the same time as slot ℓ_2. If two non-overlapping operations, such as processing on the same finishing line or a shipping event and a processing operation, are assigned to ℓ_1 and ℓ_2, then ℓ_2 must start after the end of ℓ_1. Equation (5) enforces this constraint. The duration of the intermediate slots $\ell : \ell_1 < \ell < \ell_2$ is considered in order to strengthen this constraint (Mouret et al., 2011).

$$\begin{aligned} &\sum_{o \in O} eo_{o,\ell_1,m} + \sum_{s \in S} es_{s,\ell_1} + \sum_{\substack{\ell \in L \\ \ell_1 < \ell < \ell_2}} dro_{o,\ell,m} + \sum_{\substack{\ell \in L \\ \ell_1 < \ell < \ell_2}} drs_{s,\ell,m} \leq \\ &\sum_{o \in O} sto_{o,\ell_2,m} + \sum_{s \in S} sts_{s,\ell_2} + H(1 - wo_{o,\ell_2,m} - ws_{s,\ell_2}) \end{aligned} \qquad \ell_1, \ell_2, \in L, \ell_1 < \ell_2, m \in M \qquad (5)$$

Tank transfer constraint

Equations (6) and (7) allow the transfer of a finished product to storage tank *k* from line *m* during slot ℓ, only if an order corresponding to that product is being processed, and if the tank is assigned to this product (i.e. $z_{j,k} = 1$) . In both constraints $\sum_{j \in J} d_{o,j}$ is a valid upper bound. Constraint (8) allows transfer between a line and a tank only if there is a feasible connection (e.g., a pipe) between them. Constraint (9) sets the maximum rate of transfer to tanks equal to the rate of processing at the finishing lines. Equation (10) sets the maximum amount that can be unloaded from a tank for a shipping event in a given slot. The maximum rate of unloading is set by constraint (11)

$$\sum_{k\in K} v^{in}_{o,\ell,m,k} \le wo_{o,\ell,m} \sum_{j\in J} d_{o,j} \qquad o\in O, \ell\in L, m\in M \tag{6}$$

$$v^{in}_{o,\ell,m,k} \le \sum_{j\in J} (z_{j,k} d_{o,j}) \qquad o\in O, \ell\in L, m\in M, k\in K \tag{7}$$

$$v^{in}_{o,\ell,m,k} \le tlc_{k,m} \sum_{j\in J} d_{o,j} \qquad o\in O, \ell\in L, m\in M, k\in K \tag{8}$$

$$v^{in}_{o,\ell,m,k} \le \sum_{j\in J} (\delta_{o,j} rate_{j,m} dro_{o,\ell,m}) \qquad o\in O, \ell\in L, m\in M, k\in K \tag{9}$$

$$v^{out}_{s,\ell,k} \le ws_{s,\ell} v_k \qquad s\in S, \ell\in L, k\in K \tag{10}$$

$$v^{out}_{s,\ell,k} \le sh_{s,j} drs_{s,\ell} \qquad s\in S, \ell\in L, k\in K \tag{11}$$

Material balance in storage tanks

The concept of priority slots ℓ_1, ℓ_2 where, $\ell_1 < \ell_2 \Rightarrow et_{\ell_1} \le st_{\hat{\ell}_2}$, allows the material balance at each tank to be calculated as in constraint (12).

$$lv_{\ell,k} = \sum_{m\in M}\sum_{o\in O}\sum_{\ell_1:\ell_1\le\ell} v^{in}_{o,\ell_1,m,k} - \sum_{s\in S}\sum_{\ell_1:\ell_1\le\ell} v^{out}_{s,\ell_1,k} \qquad \ell\in L, k\in K \tag{12}$$

Constraint (13) limits the level of inventory to the capacity of tank *k*,

$$lv_{\ell,k} \le v_k \qquad \ell\in L, k\in K \tag{13}$$

Tank assignment constraints

Equation (14) enforces the condition that at most one product can be assigned to each dedicated tank. Constraint (15) establishes the minimum and maximum number of tanks to which a product can be assigned. This equation can be relaxed if the user of the model is not limited by any practical constraint on the minimum or maximum number of tanks required by a product. Constraint (16) allows an assignment only if the chemical and physical properties of a product are compatible with the operating conditions of a tank.

$$\sum_j z_{j,k} \leq 1 \qquad k \in K \qquad (14)$$

$$mnk_j \leq \sum_k z_{j,k} \leq mxk_j \qquad j \in J \qquad (15)$$

$$z_{j,k} \leq cmp_{j,k} \qquad j \in J, k \in K \qquad (16)$$

Variable domain specifications

$$wo_{o,m,\ell} \in \{0,1\} \qquad o \in O, \ell \in L, m \in M \qquad (17)$$

$$ws_{s,\ell} \in \{0,1\} \qquad s \in S, \ell \in L \qquad (18)$$

$$z_{j,k} \in \{0,1\} \qquad j \in J, k \in K \qquad (19)$$

$$v^{in}_{o,\ell,m,k}, v^{out}_{s,\ell,k} \in \Re_+ \qquad o \in O, \ell \in L, m \in M, k \in K, s \in S \qquad (20)$$

$$sto_{o,\ell,m}, dro_{o,\ell,m}, eo_{o,\ell,m} \in \Re_+ \qquad o \in O, \ell \in L, m \in M \qquad (21)$$

$$sts_{s,\ell}, drs_{s,\ell}, es_{s,\ell} \in \Re_+ \qquad s \in S, \ell \in L \qquad (22)$$

$$lv_{\ell,k} \in \Re_+ \qquad \ell \in L, k \in K \qquad (23)$$

$$ap \in \Re_+ \qquad (24)$$

Example 1

In this example we wish to determine the optimal tank assignment and optimal schedule for a system of 5 tanks and 2 finishing lines where 3 products are processed for 8 orders over a 2 week time horizon. We have a set of production orders that arrive during an interval of two weeks. This set of production orders is representative of the frequency of orders and the quantity ordered during long term operation of the system. Thus, it can be used for optimally assigning tanks to products. Tables B1 – B3 in Appendix B contain the data required for the optimization model. In this example we consider that all products can be stored in all tanks, and that there is a feasible connection between every line and every tank.

The TFOP formulation described in the mathematical model section is used to solve Example 1. After some pre-analysis, 6 priority time slots are postulated. The main idea is that we need 4 slots to process the 8 orders in the 2 production lines, and, given the rates of unloading of tanks, about 2 or 3 slots for shipping operations. Computational results indicate that 6 priority slots yield the same solution as 7 but require less computational time. The resulting Mixed-Integer Linear Programming (MILP) model has 195 binary variables, 1,471 continuous variables, and 3,358 constraints. It was implemented in GAMS version 23.6 for Windows and solved using Gurobi 4.0.1 with an Intel Core i7 CPU at 2.93 GHz, and 4.00 GB of RAM. All other examples were solved with the same hardware and software. A solution within 0.2% of the optimum was found in 21 CPU seconds. The results are summarized in Table 4, and Figures 7 – 8. As seen in Table 4, 1.4 ton of product B cannot be allocated.

Table 4. Allocated product for Example 1

Product	Ordered amount [ton]	Allocated Quanity [ton]
A	215	215
B	244	242.6
C	206	206
Total	**665**	**663.6**

Figure 6 shows the optimal product-tank assignment. The total production volume specified by the production orders of Product B is the highest, and it is assigned the largest total storage tank capacity. Product A has a larger production target than product C, but product A is allotted less storage space. Figure 7 shows that the two production orders of product A are scheduled at the beginning and end of the time horizon, allowing for the complete unloading of the storage tank. This fact is observed in Figure 8(d). The Gantt chart in Figure 7 shows how the scheduling of non-overlapping operations, such as the processing operations in the same line, and processing and unloading operations are never scheduled in the same slot. From this figure we can also confirm that the ordering of the priority slots is maintained: every slot that has a lower numbering than a shipping slot ends before the shipping starts.

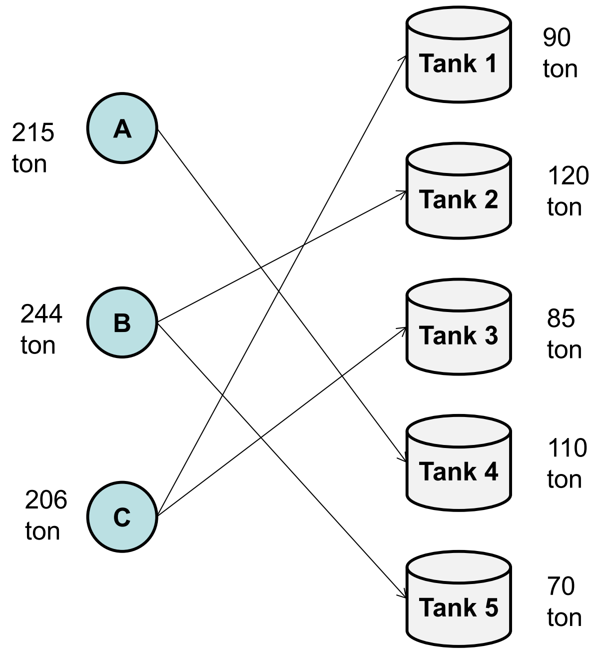

Figure 6. Optimal product-tank assignment in Example 1

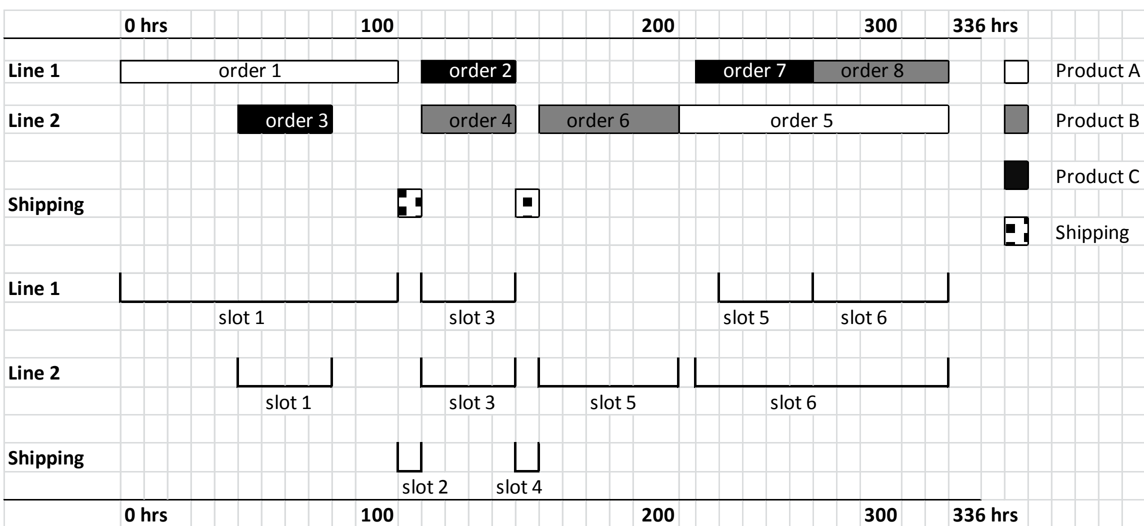

Figure 7. Gantt chart of optimal schedule in Example 1

The plots of the tank levels in Figure 8 show a common pattern. Towards the end of the operating horizon all tanks except number 5 are completely full. This result is a combination of the objective function of the TFOP that requires a maximal amount of product going into the tanks, and the finite operating horizon. In real-life operations a shipping event would probably take place at the end of the time horizon. A cyclic schedule is an approach that eliminates this effect in case it is found to be an undesirable way of operating the system. We explore this alternative in Example 3.

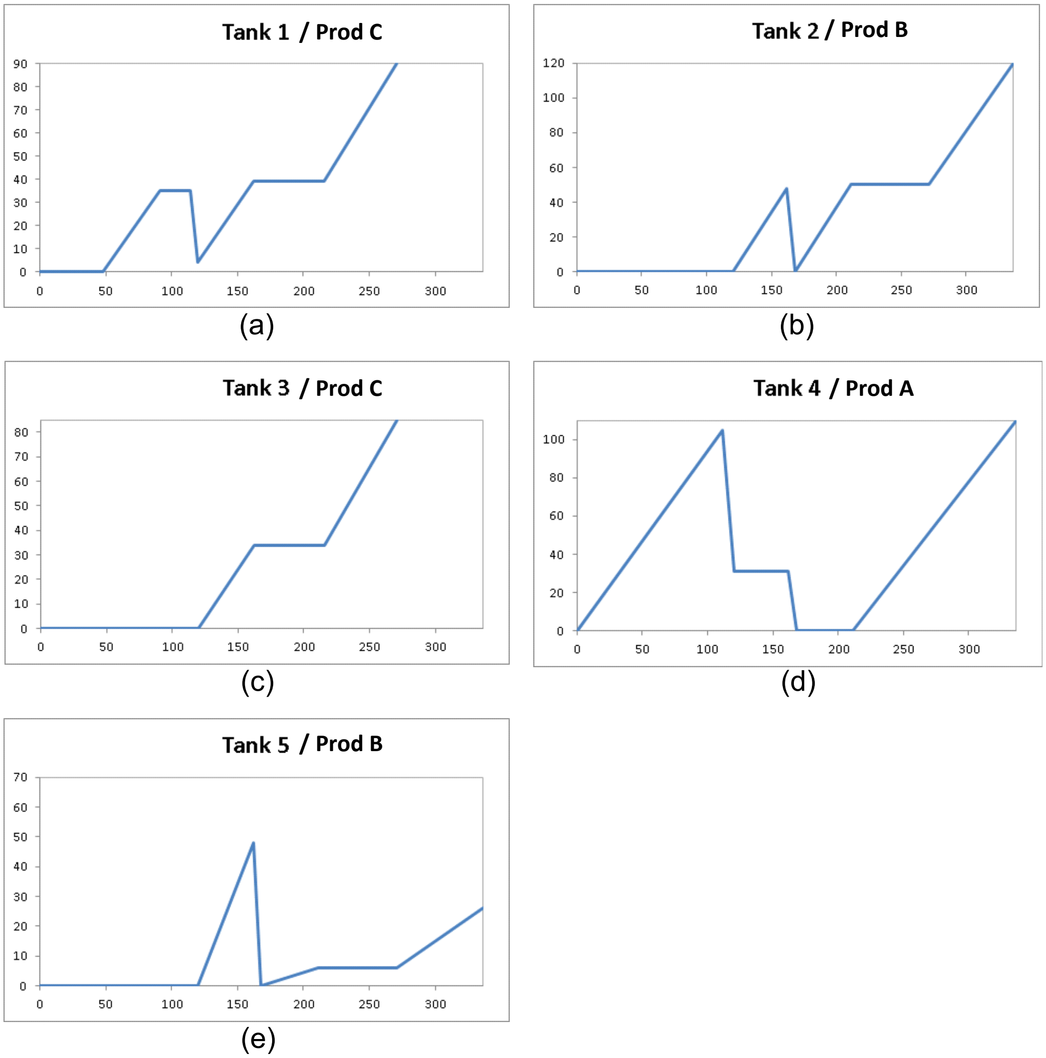

Figure 8. Tank levels in the optimal solution to Example 1

Example 2

The purpose of this example is to highlight the advantage of solving the TFOP including scheduling of processing orders instead of considering a fixed processing schedule. We consider a system of 2 parallel finishing lines where 8 products are processed for 21 orders and fed into 10 storage tanks. The operating horizon is 672 hours (4 weeks). Tables B4 – B6 contain the operating data of this system.

In the case where the schedule is fixed each production order starts exactly at the release date and lasts until it is completed or until a new order has to be processed. Every time that the tanks block the finishing lines, there is a quantity of unallocated product generated that is equal to the time the line was blocked multiplied by the processing rate of the line. The 21 production orders can in principle be processed in 11 priority slots by the two finishing lines, and we estimated 2 or 3 shipping events per week. After some computational experiments we specified 18 priority slots. These computational experiments involved starting with 15 priority slots (11 for processing the orders and 4 for one shipping event per week) and increasing the number of slots by 1 until there was no improvement in the solution. The minimum number of slots that could be used for finding this solution was 18. The resulting MILP has 1,340 binary variables, 16,561 continuous variables, and 40,261 constraints.Table 5 shows the best solutions and solution times found with fixed schedule and with optimal scheduling. Note that fixing the schedule actually makes the problem harder to solve, presumably because it becomes more constrained.

Table 5. Computational results of the TFOP with fixed and optimal scheduling

	Fixed Scheduling	**Optimal Scheduling**
Best Solution [ton]	**439**	**517**
Optimality Gap [%]	**19.9**	**1.7**
Linear Relaxation [ton]	**526**	**526**
CPU [second]	**10,000**	**530**

The total quantity required by production orders is 526 tons. The best solution with fixed schedule (439 tons) after 10,000 CPU s corresponds to approximately 16% unallocated product, while a solution of 517 tons involving less than 2% unallocated product is found when optimal scheduling is included in the TFOP. Table B4 shows that seven production orders are released between hour 0 and hour 21. A similar accumulation of orders occurs between hours 540 and 560. When the schedule requires each order to be processed at its release date, as is the case with fixed schedule, some orders have to be cut short or missed all together.

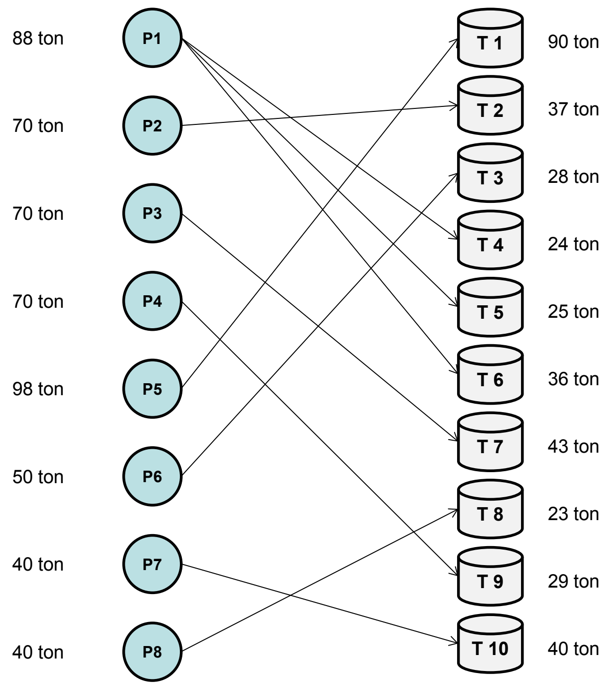

Figure 9 Optimal tank assignment with fixed schedule

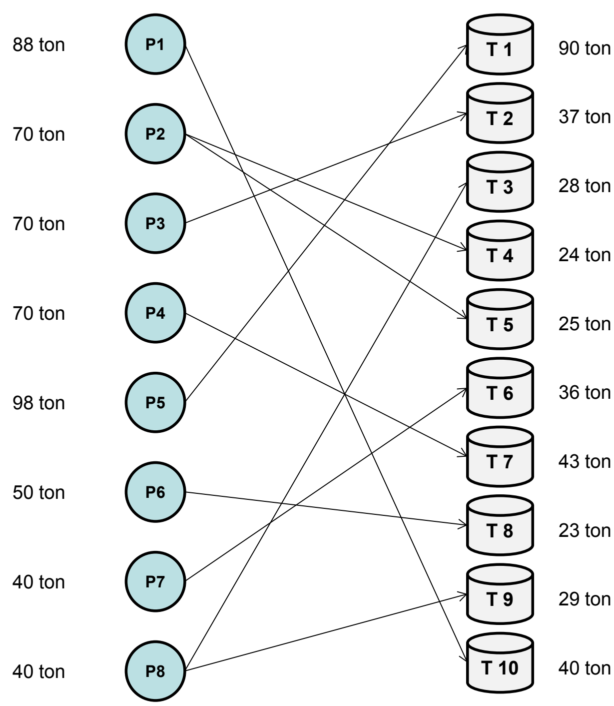

Figure 10., Optimal tank assignment with optimal schedule

The main point in this example is that including the optimal scheduling in the TFOP has a significant effect on the quality of the solution found and on the computation times required. When scheduling is included, a solution within 2% of the optimum and involving almost no unallocated product can be found in 530 CPUs. In contrast, the best solution found after 10,000 CPUs for the problem with fixed scheduling involves significant amounts of unallocated product and an optimality gap of ~20%.

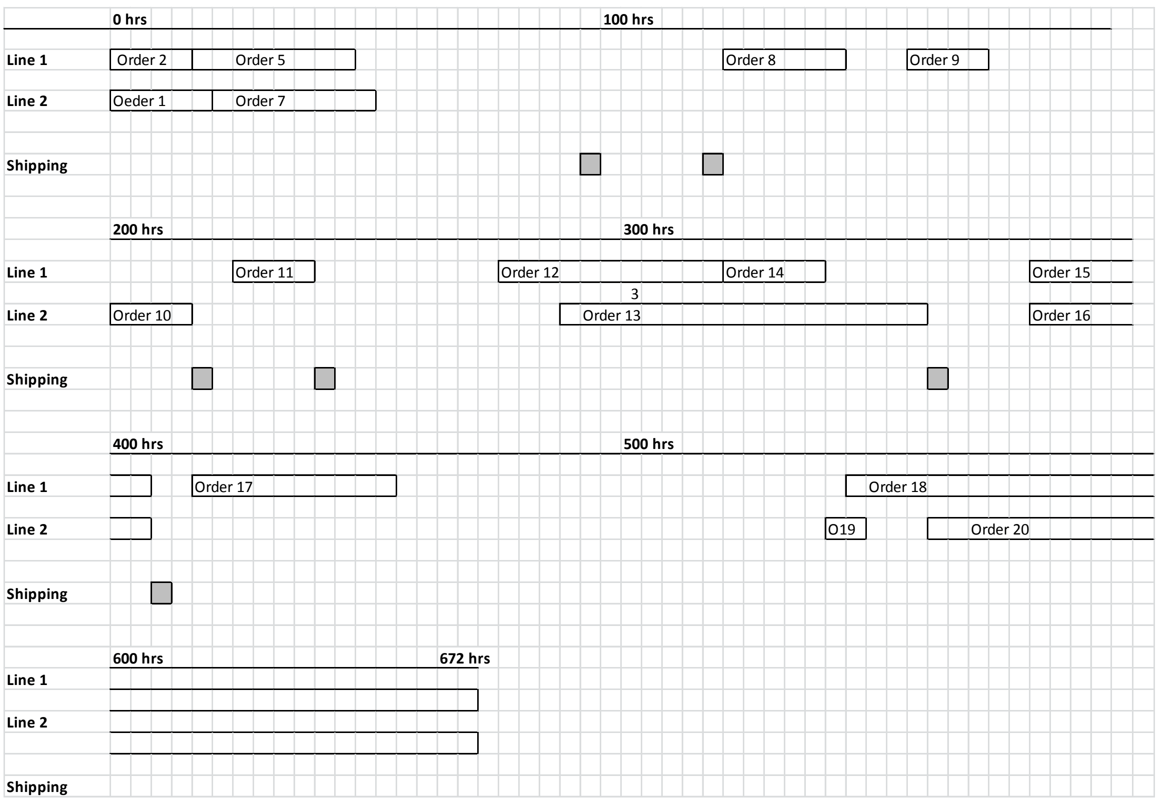

Figure 11a. Gantt Chart Example 2 with fixed schedule

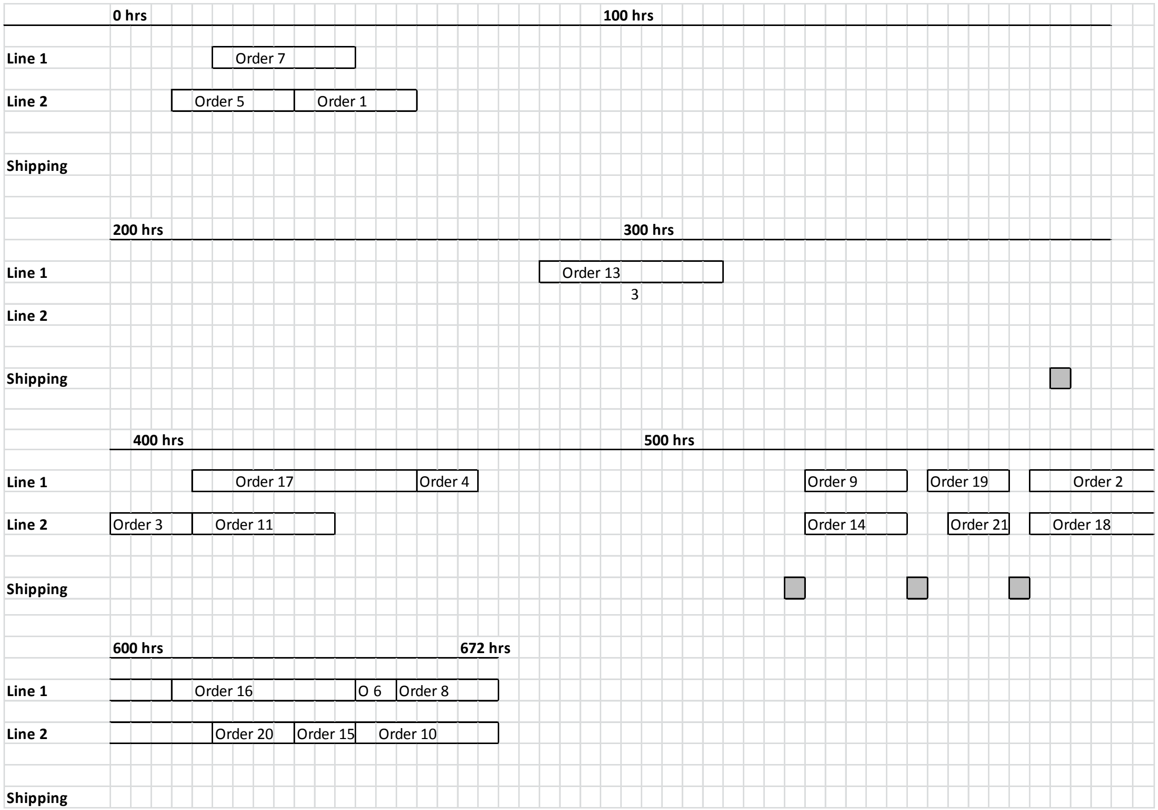

Figure 11b. Gantt Chart Example 2 with optimal schedule

Example 3

This example involves the same system as in Example 2 but with an extended time horizon of 6 weeks (1008 hrs.), which is longer than usual for scheduling problems. The reason is that we require a schedule that is representative of the medium to long-term operation of the plant in order to make the tank allocation decisions. An important difference from previous examples is that we consider a cyclic operating mode where the cycle time is 1008 hrs. A new variable, s_k^0, is introduced in the equation for the inventory balance, yielding equation (25).

$$lv_{\ell,k} = s_k^0 + \sum_{m\in M}\sum_{o\in O}\sum_{\ell_1:\ell_1\leq\ell} v_{o,\ell_1,m,k}^{in} - \sum_{s\in S}\sum_{\ell_1:\ell_1\leq\ell} v_{s,\ell_1,k}^{out} \qquad \ell\in L, k\in K \qquad (25)$$

The following constraint to enforce similar initial and final inventory is added to the model:

$$s_k^0 - \varepsilon_k \le lv_{\ell,k} \le s_k^0 + \varepsilon_k \qquad \ell = |L|, k \in K\,, \qquad (26)$$

where ε_k is a small scalar introduced to constraint (26) in order to relax the strict equality between initial and final inventory in cyclic scheduling. Computations showed that this relaxation has a significant effect on the speed of convergence of the MILP solver we used. For this case study, we set ε_k to 2. For instance, increasing the value of ε_k from 2 to 5 reduces the required CPU time to about one fourth. However, we considered a value of 2 the largest reasonable slack in equation (26).

To ensure the slot $\ell = |L|$ corresponds chronologically to the last slot, we enforce the following precedence constraint:

$$\sum_{o \in O} eo_{o,\ell_1,m} + \sum_{s \in S} es_{s,\ell_1} \le \sum_{o \in O} sto_{o,\ell_2,m} + \sum_{s \in S} sts_{s,\ell_2} \qquad \ell_1 \in L, \ell_2 = |L|, m \in M \qquad (27)$$

Table B7 contains data of the orders corresponding to the extra two weeks in Example 3. Selecting 24 priority slots using the same methodology as in Example 2, the resulting MILP has 2,500 binary variables, 31,691 continuous variables, and 77,212 constraints.

Table 6 shows the best solution found after 7 hours of computations. The product tank assignment is shown in Figure 12, while the initial and final inventories are shown in Table 7.

Table 6. Computational results of the TFOP with cyclic scheduling

Best Solution [ton]	**769**
Unallocated product [%]	**7.5**
Optimality Gap [%]	**5.9**
CPU [second]	**24,538**

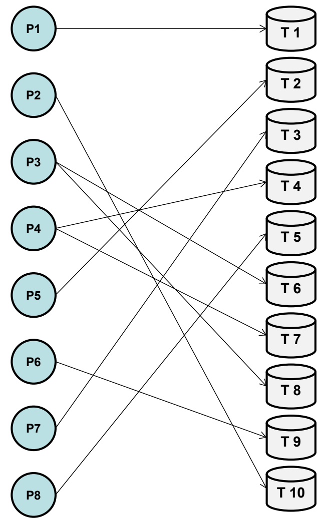

Figure 12 Tank assignment with cyclic schedule

Table 7. Initial and final inventory in storage tanks

Tank	Capacity [ton]	Initial inventory [ton]	Final inventory* [ton]
T1	90	0.0	2.0
T2	37	19.0	21.0
T3	28	0.0	2.0
T4	24	8.8	10.8
T5	25	0.0	2.0
T6	36	12.0	14.0
T7	43	27.6	29.6
T8	23	13.0	15.0
T9	29	13.3	15.3
T10	40	38.0	40.0

*Final inventory = Initial inventory +/- 2

After 7 hours of CPU time the optimality gap is still 6%. We are interested in obtaining a lower bound on the amount of unallocated product (an upper bound to equation *obj*) that would be possible to achieve if storage tank space was not a limitation. Therefore, we solve a relaxation of Example 3 that involves unlimited storage capacity.

Remark Since we allow several lines to feed the same tank simultaneously, and we assume there is at least as many tanks as products, unlimited storage capacity turns the product-assignment into a meaningless decision variable. For this reason we assign a product to a tank arbitrarily and *a priori*.

Table 8. Computational results of the TFOP with cyclic scheduling with unlimited tank sizes

Best Solution [ton]	**828**
Unallocated product [%]	**0.4**
Optimality Gap [%]	**0.4**
CPU [second]	**219**

Table 9. Storage requirements, initial, and final inventory in TFOP with unlimited tank sizes

Product	**Maximum inventory [ton]**	**Initial inventory [ton]**	**Final inventory* [ton]**
P1	**76.0**	**46.0**	**48.0**
P2	**98.0**	**58.0**	**60.0**
P3	**50.0**	**0.0**	**2.0**
P4	**65.1**	**25.1**	**27.1**
P5	**50.0**	**0.0**	**2.0**
P6	**50.0**	**0.0**	**0.0**
P7	**40.0**	**0.0**	**0.0**
P8	**25.0**	**3.0**	**5.0**

* Final inventory = Initial inventory +/- 2.

Comparing Tables 14 and 16 we can see the storage requirement for the solution corresponding to 828 tons of allocated product (vs. 769 tons in the finite storage setting) requires more storage capacity than what is available in the tank farm. The total storage capacity required to meet the maximum inventory levels obtained from solving the problem assuming unlimited storage capacity is 454 tons, while the original problem with finite storage has a total capacity of 375 tons. This type of analysis could be used to evaluate capital investment decisions in tank farms.

Conclusions and future work

We have presented a novel MILP formulation for the Tank Farm Operation Problem (TFOP) that integrates continuous production scheduling with storage resource allocation when the storage vessels are dedicated tanks. One of the examples in this paper shows the impact of including optimal scheduling in the TFOP, as opposed to assuming a fixed schedule. Even though the scheduling part of the problem corresponds to an efficient model for continuous-time scheduling based on the idea of Multi-operation Sequencing (Mouret et al., 2011), the scheduling horizon that can be contemplated within reasonable computational time is limited to a few weeks. For this reason a representative set of production orders and release dates has to be chosen in order to obtain an efficient product-assignment for medium to long-term operation. Alternatively, a cyclic schedule can be assumed as we did in Example 3.

We envision our MILP formulation combined with Discrete Event Simulation (DES) like the one in Sharda and Vazquez (2009) as part of a comprehensive decision support system. The DES model could be run by fixing the scheduling and tank farm decisions obtained by the optimization step detailed in this work. The results can be used to verify the feasibility of the optimal decisions with a simulation model that is able to capture more detailed dynamics of the problem such as simultaneous loading and unloading of storage tanks. The DES formulation mentioned before by Sharda and Vazquez (2009) is tailored specifically to this problem and was designed for industrial use. Consequently, testing and integration with this simulation tool is a natural next step. In this way, the MILP capabilities of rigorous search among alternative could be combined with the capability for representing complex operational issues of DES.

Acknowledgment

The authors acknowledge Naoko Akiya and Bikram Sharda for their contributions to defining this problem. The authors are also grateful for the financial support provided by the Dow Chemical Company.

References

Banks J., J.S. Carson II, B.L. Nelson, and D.M. Nicol (2005). Discrete event system simulations. 4th ed. Prentice Hall International series in Industrial and Systems.

Cassandras C.G. (1993). Discrete Event Systems: Modeling and Performance Analysis. Irwin Publ., 1993.

Chen, E.J., M.L. Young, and P.L. Selikson (2002). A simulation study of logistic activities in a chemical plant. *Simulation Modeling practice and Theory, 10*, 235 – 245.

Chryssolouris G., N. Papakostas, and D. Mourtzis (2005). Refinery short-term scheduling with tank farm, inventory and distillation management: An integrated simulation based approach. *European Journal of Operational Research, 166*, 812 – 827.

Erdirik-Dogan M. and I. E. Grossmann (2008). Simultaneous planning and scheduling of single-stage multiproduct continuous plants with parallel lines. *Computers and Chemical Engineering, 32*, 2664 – 2683.

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

Furman, K.C., Z. Jia, and M.G. Ierapetritou (2007). A robust event-based continuous time formulation for tank transfer scheduling. *Industrial and Engineering Chemistry Research, 46*, 9126 – 9136.

Ha J., H. Chang, E. S. Lee, In. Lee, B.S. Lee, and G. Yi (2000). Intermediate storage tank operation strategies in the production scheduling of multi-product batch process. *Computers and Chemical Engineering, 24*, 1633 – 1640.

Hvattum, L.M., K. Fagerholt, V. A. Armentano (2009). Tank allocation problems in maritime bulk shipping. *Computers and Operations Research, 36*, 3051 – 3060.

Imagine That Inc. (2007). *ExtendSim User Guide, Release 7.*

Kondili E., C.C. Pantelides, and R.W.H. Sargent (1993). A general algorithm for short-term scheduling of batch operations-I. MILP formulation. *Computers and Chemicals Engineering, 2,* 211 – 227.

Kopanos, G.M., L. Puigjaner, C. T. Maravelias (2011). Production planning and scheduling of parallel continuous processes with product families. *Industrial and Engineering Chemistry Research 50*, 1369 – 1378.

Lima, R.M., I.E. Grossmann, and Y. Jiao (2011). Long-term scheduling of a single-unit multi-product continuous process to manufacture high performance glass. *Computers and Chemical Engineering 35*, 554 – 574.

Mouret S., I.E. Grossmann, and P. Pestiaux (2009). A novel priority-based continuous-time formulation for crude-oil scheduling problems. *Industrial and Engineering Chemistry Research, 48*, 8515 – 8528.

Mouret S., I.E. Grossmann, and P. Pestiaux (2011).Time representations and mathematical models for process scheduling problems. *Computers and Chemical Engineering, 35*, 1038 – 1063.

Pantelides, C.C. (1994). Unified frameworks for process planning and scheduling. In *Foundations of computer-aided process operations*. New York: CACHE publications, pp. 253 – 274.

Shaik M.A. and C.A. Floudas (2007). Improved Unit-Specific Event-Based Continuous-Time Model for Short-Term Scheduling of Continuous Processes: Rigorous Treatment of Storage Requirements. *Industrial and Engineering Chemistry Research 46*, 1764 – 1779.

Sharda B. and A. Vazquez (2009). Evaluating Capacity and Expansion Opportunities at Tank Farm: A Decision Support System using Discrete Event Simulation. *Winter Simulation Conference 2009,* 2218 – 2224.

Stewart M.D. and L.D. Trierwiler (2005). Simulating optimal tank farm design. *PTQ magazine, Q2*, 1 – 5.

Sundaramoorthy A. and C.T. Maravelias (2008). Modeling of Storage in Batching and Scheduling of Multistage Processes. *Industrial and Engineering Chemistry Research, 47*, 6648 – 6660.

Vecchietti, A.R., J. Montagna (1998). Alternatives in the optimal allocation of intermediate storage tanks in multiproduct batch plants. *Computers and Chemical Engineering, 22(1)*, S801 – S804.

Zeng Q. and Z. Yang (2009). Integrating simulation and optimization to schedule loading operations in container terminals. *Computers and Operations Research, 36,* 1935 – 1944.

Appendix A: Nomenclature

Sets/Index

J / j	Products
K / k	Tanks
L / ℓ	Time slots
M / m	Finishing lines
O / o	Production orders
S / s	Shipping or unloading events

Variables

ap	total amount of finished product allocated to storage tanks
$dro_{o,m,\ell}$	duration of processing of order o in line m during slot ℓ
$drs_{s,\ell}$	duration of shipping event s in slot ℓ
$eo_{o,m,\ell}$	end time for processing order o in line m during slot ℓ
$es_{s,\ell}$	end time of shipping event s in slot ℓ
$lv_{\ell,k}$	inventory levels in tank k at the end of slot ℓ
s_k^0	initial inventory in tank k
$sto_{o,m,\ell}$	start time for processing order o in line m during slot ℓ
$sts_{s,\ell}$	start time shipping event s in slot ℓ
$v_{o,\ell,m,k}^{in}$	quantity of material corresponding to processing order o transferred to tank k from line m during slot ℓ
$v_{s,\ell,k}^{out}$	quantity of material shipped or unloaded from tank k during unloading event s in slot ℓ

$wo_{o,m,\ell}$ — binary variable to denote that order o is processed in line m during slot ℓ

$ws_{s,\ell}$ — binary variable to denote that unloading event s occurs in slot ℓ

$z_{j,k}$ — binary variable to denote that tank k is dedicated to product j

Parameters

$cmp_{j,k}$ — 1 if physical and chemical properties of product j are compatible with the operating conditions of tank k; 0 otherwise

$d_{o,j}$ — amount of finished product j requested by processing order o

$\delta_{o,j}$ — 1 if product j is requested in processing order o; 0 otherwise

H — length of time horizon

mxk_j — maximum number of tanks that can be assigned to product j

mnk_j — minimum number of tanks that must be assigned to product j

$rate_{j,m}$ — rate of production of product j in line m

rd_o — release date of processing order o

$shipwin_s$ — time window during which unloading of tank into modes transportation is allowed; the length of this shipping window is predetermined by the logistics of the process system

$sh_{s,j}$ — shipping rate of product j in shipping event s

shd_s — maximum duration of shipping event s

$stime_s$	time when a shipping event s is allowed to start; the maximum shipping event frequency is determined a priori, according to logistics of modes of transportation
$tlc_{l,k}$	1 if any product can be transferred between line ℓ and tank k; 0 otherwise
v_k	volume of tank k
ω_o	Weighting parameter of order o according to its importance in the production schedule

Appendix B: Data Tables

Table B1. Production orders in Example 1

Order	Product	Quantity [ton/hr]	Release date [hr]
1	A	105	0
2	C	69	0
3	C	35	48
4	B	98	72
5	A	110	96
6	B	56	168
7	C	102	216
8	B	90	264

Table B2. Production rate and tank capacities in Example 1

Production rate [ton/hr]				Tank	Capacity [ton]
	Line 1	Line 2			
A	0.95	0.89		1	90
B	1.09	1.15		2	120
C	0.92	0.82		3	85
				4	110
				5	70

Table B3. Interval between shipping, duration of unloading, and unloading rate

Tank	Interval [hr]	Duration [hr]	Transfer rate [ton/hr]
1	24	6	12.07
2	24	6	13.86
3	24	6	12.01
4	24	6	12.31
5	24	6	11.83

Table B4. Production orders in Example 2

Order	Product	Quantity [ton]	Release date [hr]
1	P1	30	0
2	P1	40	0
3	P1	18	10
4	P2	10	14
5	P2	40	16
6	P2	20	18
7	P3	30	21
8	P3	20	124
9	P3	20	156
10	P4	30	198
11	P4	30	220
12	P4	10	272
13	P5	50	284
14	P5	30	316
15	P5	18	378
16	P6	25	380
17	P6	25	412
18	P7	30	544
19	P7	10	536
20	P8	25	558
21	P8	15	560

Table B5. Production rate and tank capacities in Example 2

Production rate [ton/hr]		
	Line 1	**Line 2**
P1	1.29	1.29
P2	1.07	1.07
P3	1.07	1.07
P4	1.07	1.07
P5	1.64	1.64
P6	0.55	0.55
P7	0.71	0.71
P8	0.71	0.71

Tank	**Capacity** [ton]
T1	30
T2	27
T3	18
T4	15
T5	15
T6	15
T7	12
T8	12
T9	15
T10	39

Table B6. Interval between shipping, duration of unloading, and unloading rate

Tank	**Interval** [hr]	**Duration** [hr]	**Transfer rate** [ton/hr]
1	24	4	2.50
2	24	4	2.25
3	24	4	1.50
4	24	4	1.25
5	24	4	1.25
6	24	4	1.25
7	24	4	1.00
8	24	4	1.00
9	24	6	0.83
10	24	6	2.17

Table B7. Processing orders in weeks 5 and 6 in Example 3

Orders	**Product**	**Quantity** [ton]	**Release date** [hr]
22	P3	30	650
23	P3	20	700
24	P3	20	710
25	P4	50	712
26	P4	30	714
27	P4	18	750
28	P1	30	770
29	P1	40	790
30	P1	18	800
31	P2	10	850
32	P2	40	900
33	P2	20	950