

New General Discrete-Time Scheduling Model for Multipurpose Batch Plants

Samuel Moniz,^{†,§} Ana Paula Barbosa-Póvoa,^{*,†} and Jorge Pinho de Sousa^{‡,§}

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

[‡]INESC TEC, 4200-465 Porto, Portugal

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

Supporting Information

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

1. INTRODUCTION

In the past decades many optimization approaches have been developed to address supply chain, planning, and scheduling problems. These developments are being motivated by the need that industries have to reduce costs, increase revenues, and, in general, to operate more efficiently. Consequently, the existing gap between theoretical optimization models and real world scheduling problems is gradually decreasing. This can be somehow justified by the increasing number of works published in the recent years that address practical optimization problems. According to Grossmann,¹ process industries are actively looking for optimization approaches that can be integrated in key decision-making processes so as to minimize costs and maximize income, while increasing the system responsiveness. In the particular case of the pharmaceutical industry, Varma et al.² argue on the importance of developing models that integrate decision-making processes related to R&D, manufacturing, supply chain, and marketing. An extensive review on the modeling approaches for scheduling problems that tackle these issues is available in the paper written by Mendez et al.,³ where the characteristics, advantages, and disadvantages of the models are deeply addressed.

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

illustrative problem instance, used to assess the developed models.

The rest of the paper is structured as follows. In Section 2, we describe an example to illustrate the impact of the definition of lots in the production schedule, and in Section 3, a literature review is presented. The problem statement is introduced in Section 4, and it is followed by the mathematical formulations in Section 5. Then in Section 6, we propose several model extensions, and in Section 7, we compare the model performances. Finally, Section 8 provides some concluding remarks.

2. ILLUSTRATIVE EXAMPLE

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

Consider the determination of a production schedule for three products: PA, PB, and PC. Task sequences and respective alternative units are depicted in Figure 1. Products PA and PB are produced from raw materials, while product PC is produced from PA and PB. The objective is to maximize the overall profit by determining a schedule that keeps record of the production lots and involves sequence-dependent changeovers between products and lots.

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

Received: July 4, 2013

Revised: November 8, 2013

Accepted: November 8, 2013

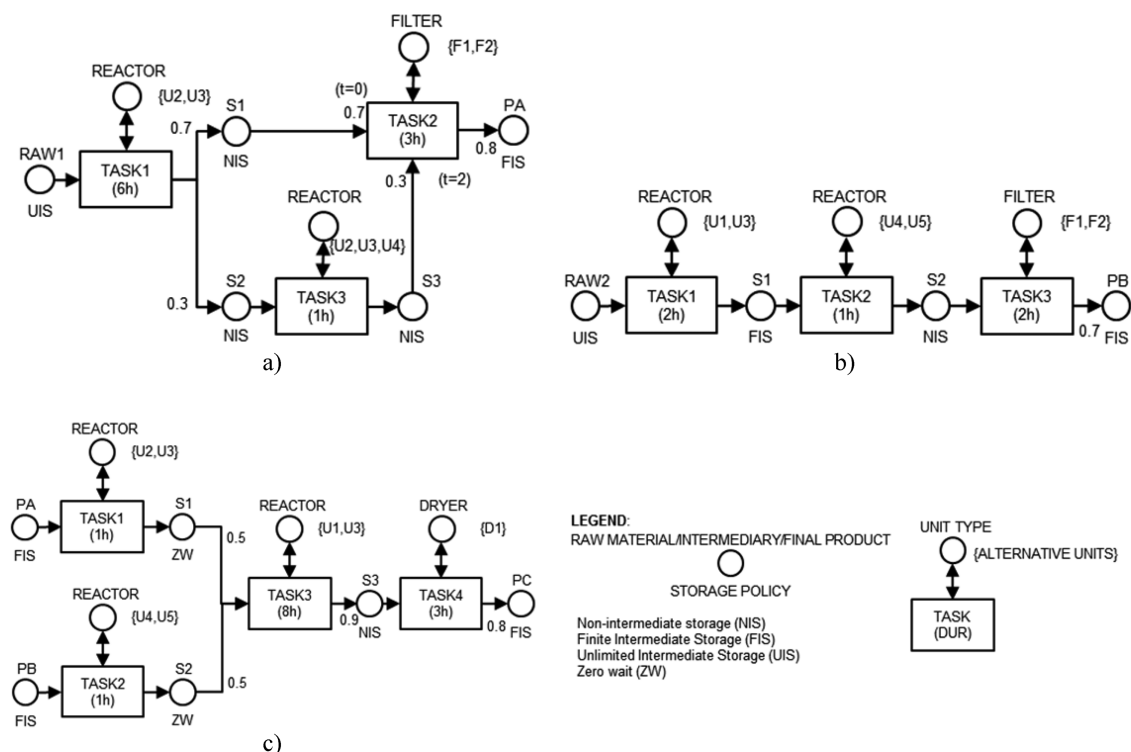


Figure 1. Illustrative example.

produced by each task that is limited by the capacity of the processing unit and is part of the production of a lot.

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

In order to illustrate the impact of production lots on scheduling, we consider a small instance with product PB for a scheduling horizon of 10 h. We want to determine a production schedule in which the task-unit assignment accounts for a given lot size and that the lots traceability is ensured. For that, we define a fixed demand equal to 3000 kg that is produced assuming two scenarios. The first is a base scenario where no lots are defined, while in the second scenario we assume two lots of 1500 kg.

In Figure 2a, we show a schedule for the base scenario, and as it can be seen, the tasks batch size is as large as possible, so as to minimize the number of tasks and therefore the production costs. The amount of material produced by two tasks TASK1 is split by three tasks TASK2 and three tasks TASK3. Since lots were not explicitly modeled, it is not possible to make a task–lot assignment; thus, the schedule of Figure 2a does not account for lot traceability.

On the contrary, the schedule depicted in Figure 2b results in the same amount of final product but considers lots traceability. The difference is in the number and respective batch sizes of the tasks. To consider lots traceability the schedule must have unique associations between tasks and lots. In our example, it can be seen that the first task TASK1 and the two first tasks TASK2 and TASK 3 are associated to lot L1, while the other tasks are

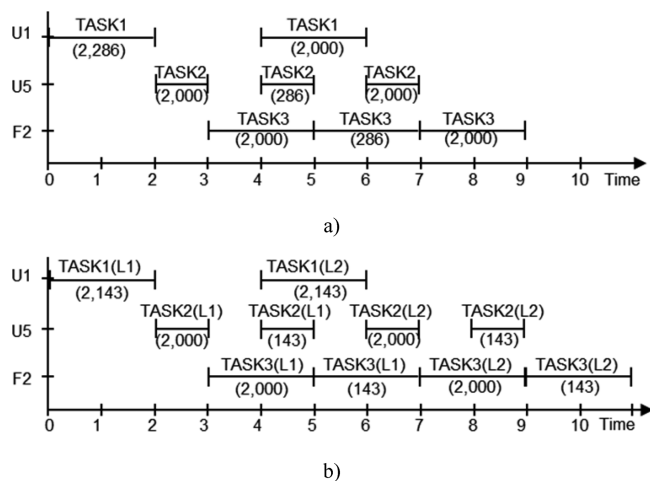


Figure 2. (a) Schedule assuming a demand of 3000 kg with no defined lots; (b) schedule with two lots of 1500 kg.

associated to lot L2. In this way, raw materials, intermediaries, and final products are distinctively associated to each lot. The impact of lots in scheduling would be higher if sequence-dependent changeovers were considered.

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

3. BACKGROUND

Scheduling of process plants has received considerable attention in the literature, with some relevant reviews on the topic.^{3–8}

Scheduling problems can be classified in terms of the network of processing tasks.³ The allowed material flow and unit specific constraints strongly determine the modeling approach and,

consequently, the model performance and its complexity. In general, we may have sequential or network processes.

In sequential processes the batch entity is preserved by ensuring that the output of a batch is consumed by a single task and the input of a batch is produced by a single batch. Within the sequential processes, single and multiple stage topologies can be defined. The former consists in production systems with just one stage and may have parallel units, and the latter involves production systems with more than one stage that may also have parallel units. Sequential processes can either use precedence-based or time-grid formulations. Precedence-based models have been proposed by several authors^{9–12} and time-grid models for sequential processes rely on time-slots.^{13–15}

On the contrary, network processes have an arbitrary topology and are usually more complex than sequential processes, since they deal with batch mixing and splitting and cyclic material flows. For these reasons, models for network topologies require resource balance constraints and are time-grid based, either discrete-time or continuous-time. By definition models used for network processes can also be applied to sequential processes, since they can model all types of process configurations. Continuous-time formulations may rely on unit specific events^{16–19} or on global events.^{20–24} The major advantage of the continuous-time formulations is that tasks may occur anywhere in the scheduling horizon and thus these models are considered more accurate. However, in terms of mathematical programming, continuous-time models generally result in large integrality gaps that tend to deteriorate computational times.

Discrete-time formulations assume that the scheduling horizon has been divided into a finite number of time intervals of fixed and equal duration. Tasks are allowed to take place just in the boundaries of the time intervals, which makes it easier to model inventory and units availability constraints. These models deal easily with material balances and inventory costs and multiple delivery dates and result into compact formulations. On the other hand, they present some difficulties when modeling variable processing times and sequence-dependent changeovers. Moreover, we need to be aware of the trade-offs between accuracy of the scheduling solutions, the time discretization, and the scheduling horizon, since computational performance strongly depends on the number of time intervals considered. Both the State-Task Network (STN) representation suggested by Kondili et al.²⁵ and Shah et al.²⁶ and the RTN representation introduced by Pantelides²⁷ have been widely used for modeling schedule problems. For example, Barbosa-Povoa and Macchietto²⁸ developed the Maximal State-Task Network (m-STN) representation that simultaneously considers operational and design characteristics. Pinto et al.²⁹ modified RTN to address design and retrofit of batch plants with periodic mode operation. Castro et al.³⁰ solved an industrial scheduling problem from the chemical–pharmaceutical industry by proposing a periodic RTN formulation. Wassick and Ferrio³¹ proposed some extensions for RTN. Sundaramoorthy and Maravelias³² developed a scheduling framework that addresses the recipes structure in network and sequential subsystems. And more recently, Moniz et al.³³ proposed a sequential approach for the simultaneous scheduling of regular and nonregular products in multipurpose-batch plants. The integrated approach is based on RTN and is applied to a real scheduling problem from the chemical–pharmaceutical industry. For a comparison of discrete-time and continuous-time models, see refs 34 and 35.

4. PROBLEM STATEMENT

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

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

5. MATHEMATICAL FORMULATIONS

5.1. Concepts and Notation. In order to compare the effectiveness of the proposed formulation (denoted later in this work by model M2), we present an additional mathematical formulation (model M1) based on the RTN formulation of Pantelides,²⁷ where scheduling aspects studied by other authors are incorporated in an integrated form. Variations of M1 formulation, in their discrete-time form, have been extensively used by other authors such as Castro et al.,³⁶ Castro et al.,³⁰ and Wassick and Ferrio.³¹

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

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

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

5.2. RTN Model (M1). We use a RTN discrete-time formulation as basis for comparison with the model proposed in this paper. Model M1 extends the RTN model of Pantelides²⁷ by considering the temporary storage in the processing units constraints defined by Kondili et al.,²⁵ the changeover variables proposed by Castro et al.,³⁰ and the multiproduct delivery extensions developed by Wassick and Ferrio.³¹ Moreover, in Section 6, we also propose some extensions to address the start of changeovers tasks, nonpreemptive lots, lots start and sizes, task-unit-layout assignment, and alternative task-unit assignment.

We assume a scheduling horizon having a length equal to T and divided into time intervals of fixed length. The model

considers the following decision variables that are defined for each time interval $t \in H$.

- The assignment of tasks to processing units decisions is done by the N_{klt} binary variables that are equal to 1 if task k starts lot l at time interval t .
- The task batch size decisions are done through the ξ_{klt} continuous variables that define the batch size of task k and lot l at time interval t .
- Changeover tasks are defined by the binary variables $C_{rll't}$ that are equal to 1 if a changeover task occurs on resource (processing unit) r between lots l and l' at time interval t .
- Resources availability is given by the R_{rlt} continuous variables that define the resource availability r at lot l and at time interval t .
- Deliveries are modeled by the Π_{rlt} continuous variables that define the delivery of resource (final products) r of lot l at time interval t . If the minimum demand is not fulfilled, then the Π_{rlt}^{slack} continuous variables will have a value equal to the amount that was not delivered.

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

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

5.3. Proposed Model (M2). Discrete-time models efficiently deal with resources balances, multiple delivery dates, and inventory costs. However, the model size significantly increases and the computational performance is seriously affected when modeling variable processing times, temporary storage in the processing units, and sequence-dependent changeovers.

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

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

In order to avoid increasing the number of binary variables of the model, as a result of modeling temporary storage and

changeovers, we have developed a new discrete-time formulation. The developed model also addresses lots blending and traceability features.

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

Figure 3 shows the conceptual differences between models M1 and M2 for the resource availability variables. While in M1 all

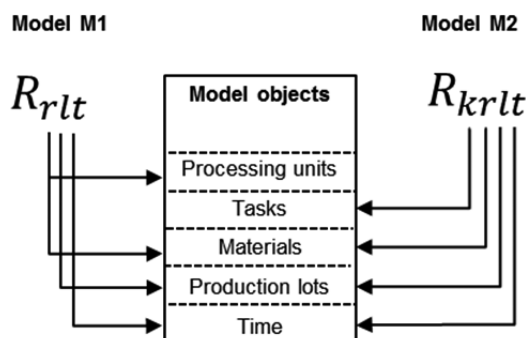


Figure 3. Resource availability variables for models M1 and M2.

resources are treated uniformly through the continuous variables R_{rlt} in M2 the continuous variables R_{krlt} define the amount of resource r (intermediaries or final products) available at time interval t and produced by task k of lot l .

The relations between products, lots, tasks, and units sets are illustrated in Figure 4. We assume that we have a set of products

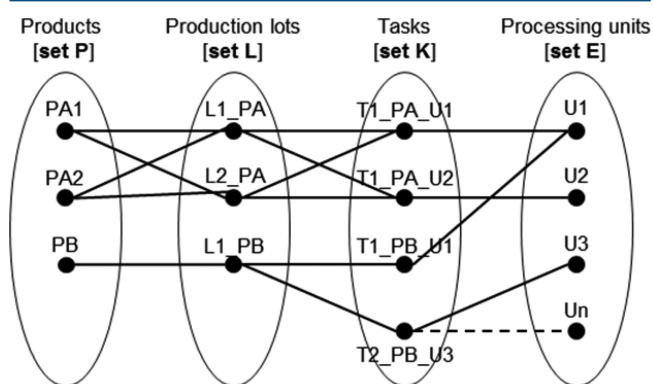


Figure 4. Relation between products, lots, tasks, and units sets.

P ; in the example we have $\{PA1, PA2, PB\}$, associated with recipes that describe the tasks sequence, the task-unit suitability, the materials needs, and the storage policies. A recipe may involve the production of one or more products. In the example shown in Figure 4, products PA1 and PA2 are subproducts of a unique recipe. Each product has at least one lot belonging to set L . Production tasks are associated to processing units and belong to set K and may execute any lot of the corresponding product. Finally, processing units belong to set E and are associated to different tasks, since they operate in a multipurpose way.

Model M2 is defined by task-unit assignment/sequencing constraints 1; materials produced and consumed, constraints 2

and 3 respectively; products blending constraints 4, materials balance constraints 5, minimum and maximum materials availability constraints 6; minimum and maximum task batch size constraints 7; demand constraints 8; delivery constraints 9 and 10; temporary storage in the processing units without or with changeovers, constraints 11 and 12, respectively; tasks started must end in the time horizon constraints 13 and variables domain constraints 14.

Constraints.

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

$$R_{krlt}^p = \sum_{\theta=0}^{\tau_k} (\nu_{kr\theta}^p \xi_{kl,t-\theta}) \quad \forall r \in I \cup P, k \in K_r^p, l \in L_r, t \in H \quad (2)$$

$$\sum_{k \in K_r^c} R_{krlt}^c = \sum_{k \in K_r^c} \sum_{\theta=0}^{\tau_k} (\nu_{kr\theta}^c \xi_{kl,t-\theta}) \quad \forall r \in I \cup P \setminus B, l \in L_r, t \in H \quad (3)$$

$$\sum_{k \in K_r^p} \sum_{l \in B_r} R_{krlt}^c = \sum_{k \in K_r^c} \sum_{l \in L_k} \sum_{\theta=0}^{\tau_k} (\nu_{kr\theta}^c \xi_{kl,t-\theta}) \quad \forall r \in B, t \in H \quad (4)$$

$$R_{krlt} = (R_{krl}^{\text{init}(m)}|_{t=0}, R_{krl,t-1}|_{t>0}) + R_{krlt}^p - R_{krlt}^c + \Pi_{krlt} \quad \forall r \in I \cup P, k \in K_r^p, l \in L_r, t \in H \quad (5)$$

$$0 \leq \sum_{k \in K_r^p} \sum_{l \in L_r} R_{krlt} \leq R_{rt}^{\max} \quad \forall r \in I \setminus \text{NIS} \cup P, t \in H \quad (6)$$

$$V_{krl}^{\min} N_{klt} \leq \xi_{klt} \leq V_{krl}^{\max} N_{klt} \quad \forall r \in E, k \in K_r, l \in L_k, t \in H \quad (7)$$

$$Q_{rld}^{\min} - \Pi_{rld}^{\text{slack}} \leq \sum_{k \in K_r^p} \sum_{t \in DW_{rld}} (-\Pi_{krlt}) \leq Q_{rld}^{\max} \quad \forall r \in P, l \in L_r, d \in D_r$$

$$DW_{rld} = \{t | r \in P, l \in L_r, d \in D_r, t \in H: T_{rd}^{dd} \geq t \geq T_{rd}^{ed}\} \quad (8)$$

$$\Pi_{krlt} = 0 \quad \forall r \in P, l \in L_r, d \in D_r, k \in K_r^p, t \in H \setminus DW_{rld} \quad (9)$$

$$\Pi_{krlt} = 0 \quad \forall r \in I, l \in L_r, k \in K_r^p, t \in H \quad (10)$$

$$\sum_{k \in K_r} \sum_{l \in L_r} N_{klt} + \sum_{k \in K_r^p} \sum_{r' \in I_k^{\text{NIS}}} \sum_{l \in L_r} \left(\frac{R_{kr'l}^p}{V_{kr'l}^{\max}} \right) \leq 1 \quad \forall r \in E, t \in H \quad (11)$$

$$\sum_{k \in I_1^r} N_{klt} + \sum_{k' \in I_1^{r'}} N_{k'l't-\tau_k-\theta+1} + \sum_{k'' \in I_1^{r''}} \sum_{r' \in I_k^{\text{NIS}}} \left(\frac{R_{k''r'l'-\theta}}{V_{k''r'l'}^{\max}} \right) \leq 1 \quad \forall r \in E, l, l' \in L_r, \theta = 0, \dots, c_{rl'l}, t \in H \quad (12)$$

$$\sum_{t=T-\tau_k+1}^T N_{klt} = 0 \quad \forall k \in K, l \in L_k \quad (13)$$

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

$$\xi_{klt} \in \mathbb{R}_+ \quad \forall r \in E, k \in K_r, l \in L_k, t \in H$$

$$\Pi_{rld}^{\text{slack}} \in \mathbb{R}_+ \quad \forall r \in P, l \in L_r, d \in D_r$$

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

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

(14)

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

Materials production R_{krlt}^p and consumption R_{krlt}^c are defined separately to address lots blending. Constraints 2 define the amount of resource r (intermediaries or final products) produced by task k of lot l at time interval t . Parameters $\nu_{kr\theta}^p$ give the production proportion of the batch size of task k for resource r . Constraints 3 give the amount of resource r consumed by task k of lot l at time interval t at the proportion $\nu_{kr\theta}^c$ of the batch size ξ_{klt} . Since resource r of lot l can be available from any tasks $k \in K_r^p$ that have produced r , the summation over R_{krlt}^c in the left-hand side of constraints 3 is required.

Constraints 4 define the special case of lots blending. In many situations, it is common to produce several lots of stable intermediaries that are used to produce other lots of final products. In these cases, blending of lots is allowed but it is necessary to ensure traceability, which is done by constraints 4. These constraints are defined for the set of intermediaries/products B whose lots can be blended.

Constraints 5 express the material r balance for each task k and lot l by considering the material in the previous time interval, the amount produced and consumed, and the material deliveries. Constraints 6 define the minimum and maximum materials/lots availability allowed for each time interval. Constraints 7 impose the task-batch size limits.

Constraints 8 define multiple product/lot deliveries Π_{krlt} for a given delivery time window DW_{rld} . The amount of resource r of lot l at delivery d is limited by the minimum Q_{rld}^{\min} and maximum Q_{rld}^{\max} quantities. Production requirements are modeled as “soft-constraints” so as to avoid infeasible solutions. Thus, missing deliveries are expressed by the continuous variables Π_{rld}^{slack} and are penalized in the objective function through coefficient $c_{rl'd}^{\text{slack}}$. Constraints 9 and 10 express the fact that delivery variables Π_{krlt} cannot take values for the time intervals out of the delivery time window and for other resources than final products.

Constraints 11 define the temporary storage in the processing units and state that if the binary variable N_{klt} is equal to 1, then R_{krlt} must be equal to 0. In other words, no task k of any lot l can start in the processing unit r if unit r is temporarily storing material from any other task. Note that the second term of the left-hand side only occurs for tasks that produce intermediaries subject to the Non-Intermediate Storage (NIS) policy, defined by the set I_k^{NIS} . Constraints 12 extend constraints 11 to account for sequence-dependent changeovers. In this way, tasks must respect the sequence-dependent changeover time defined for each unit and lot by the parameter $c_{rl'l'}$ and for possible storage time in the processing units. Therefore, if task k of lot l occurs at time t , then the first term of the constraint is equal to one, and the second and third terms are forced to be zero for all tasks k' and k''

belonging to lots l' and for the time intervals corresponding to $t - \tau_k - \theta + 1$ for the production tasks and to $t - \theta$ for the temporary storage.

Constraints 13 define that tasks must finish in the time horizon of interest and constraints 14 state the non-negativity of the continuous variables resource availability, production and consumption, batch size, and missing delivery; the nonpositivity of the delivery variables; and the integrality of the assignment/sequencing variables.

5.4. Objective Function. The objective is to maximize the economical result of the global operation (see expression (15)) by taking into account the value of the products (VP), the storage costs (SC), the operational costs (OC), and the missing deliveries costs (MC). Note that model M2 cannot take into account changeover costs since there are no changeover variables, and in order to make a fair comparison between models M1 and M2, changeover costs were not considered in the objective function.

Objective Function.

$$\begin{aligned} \max Z = & \text{value of the products (VP)} - \text{storage costs (SC)} \\ & - \text{operational costs (OC)} \\ & - \text{missing delivery costs (MC)} \end{aligned} \quad (15)$$

$$VP = \sum_{r \in P} \sum_{l \in L_r} \sum_{k \in K_r^p} \sum_{t \in H} (-v_r \Pi_{krlt}) \quad (15a)$$

$$SC(M1) = \sum_{r \in I \cup P} \sum_{l \in L_r} \sum_{k \in K_r^{sto}} \sum_{t \in H} (c_r^{sto} (\xi_{klt} + R_{rlt})) \quad (15b)$$

$$SC(M2) = \sum_{r \in I \cup P} \sum_{l \in L_r} \sum_{k \in K_r} \sum_{t \in H} (c_r^{sto} R_{krlt}) \quad (15c)$$

$$OC = \sum_{k \in K_r} \sum_{l \in L_r} \sum_{t \in H} (c_k^{op} N_{klt}) \quad (15d)$$

$$MC = \sum_{r \in P} \sum_{l \in L_r} \sum_{d \in D_r} (c_{rld}^{slack} \Pi_{rld}^{slack}) \quad (15e)$$

The first term of the objective function defines the value of the delivered products; see expression 15a. The second term determines the storage costs, which are calculated differently for models M1 and M2. So, for model M1 storage costs are associated to materials stored under FIS and UIS policies that can be expressed by the continuous R_{rlt} variables and by materials temporarily held by the processing units; see expression 15b. For model M2, storage costs are determined simply by the continuous variables R_{krlt} since the availability of the materials is only modeled through these variables; see expression 15c. The fourth term of the objective function determines the operational costs; see expression 15d. And the fifth term is a penalty cost associated with the missing deliveries; see expression 15e.

6. MODELS EXTENSIONS

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

Changeovers Start (M1). In model M1, as stated by constraints A1, changeover tasks may occur in any time interval between the start of the tasks associated to lots l and l' . However, a common industrial practice is to perform the changeover as

soon as the task finishes. We illustrate this situation in Figure 5, with (a) showing the time range where the changeover tasks may

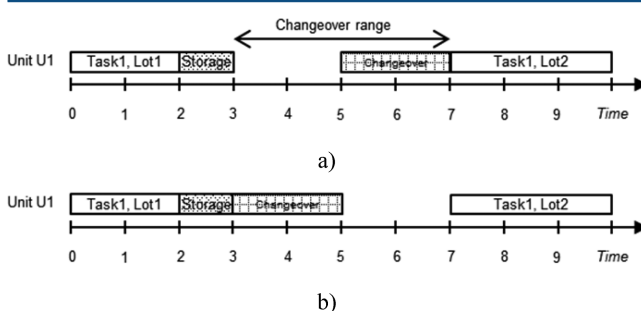


Figure 5. Start of production, storage, and changeover tasks (Model M1).

occur if constraints A1 are used. However, the desirable scheduling solution is the one presented in (b), since the changeover occurs immediately after the storage tasks.

Constraints 16 force changeovers to occur immediately after a production or storage task. Another relevant point is that constraints 16 help in reducing the model degeneracy.

$$\sum_{k \in K_r} N_{kl, t-\tau_k} - \sum_{l' \in L_r} C_{rll't} \geq 0 \quad \forall r \in E, l \in L_r, t \in H \quad (16)$$

Non-Preemptive Lots (M1). It is also a common practice in many chemical batch plants that lots once started in one unit cannot be interrupted to allow the production of a different lot. Constraints 17 define that if a changeover from lot l' to lot l occurs in unit r , then no changeover can occur in that unit from l to l' .

$$\sum_{l' \in L_r} C_{rll't} + \sum_{l' \in L_r} C_{rll't} \leq 1 \quad \forall r \in E, l \in L_r \quad (17)$$

Lots Start (M1 and M2). Constraints 18 state that lot l is only executed if the previous $l-1$ is also executed. Thus, if task k of lot l is performed at time t , then the same task k of lot $l-1$ should have started previously or any alternative tasks A_k to k should have started at the time intervals between $t' = 0$ and t . This allows different lots to be produced in parallel.

$$\begin{aligned} N_{klt} - \sum_{t'=0}^{t-\tau_k} N_{k, l-1, t'} - \sum_{k' \in A_k} \sum_{t'=0}^t N_{k', l-1, t'} \leq 0 \quad \forall \\ k \in K, l \in L_k, t \in H: l > 1 \end{aligned} \quad (18)$$

Lots Sizes (M1 and M2). If we want to define lots with exactly the same amount of material, constraints 19 and 20 may be applied. Constraints 19 impose that the total amount produced by tasks of different lots must be the same and constraints 20 state that the number of tasks must be the same among the lots.

$$\sum_{t \in H} \xi_{klt} = \sum_{t \in H} \xi_{kl't} \quad \forall k \in K, l, l' \in L_k, l \neq l', t \in H \quad (19)$$

$$\sum_{t \in H} N_{klt} = \sum_{t \in H} N_{kl't} \quad \forall k \in K, l, l' \in L_k, l \neq l', t \in H \quad (20)$$

Task-Unit-Layout Assignment (M1 and M2). For processes with many alternative units it may be preferable to

do the task-unit assignment taking into consideration the physical layout of the units. Figure 6 depicts the plant layout

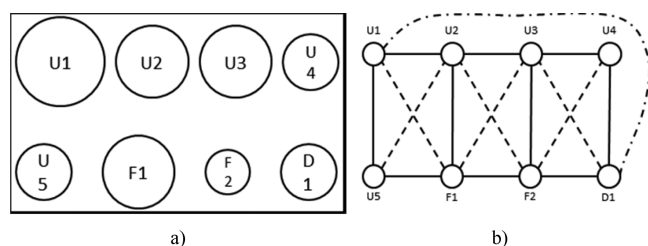


Figure 6. Location of processing units: (a) plant layout; (b) allowable connection between units.

and the allowed connections between units for the processes of Figure 1. For example, unit U1 can only transfer/receive materials to/from U2, U5, F1, and also D1.

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

To model this requirement, we have created new binary variables X_{kl} that are equal to 1 if task k is assigned to lot l , see expression 21.

$$X_{kl} = \begin{cases} 1 & \text{if task } k \text{ is assigned to lot } l \\ 0 & \text{otherwise} \end{cases} \quad \forall k \in K, l \in L_k \quad (21)$$

If task k and k' use processing units that are connected, then $(k, k') \in S$. Constraints 22 define that if task k is assigned to lot l , then task k' cannot be assigned to the same lot, since k is not connected to k' . Constraints 23 ensure that if $\sum_{t \in H} N_{klt} > 0$ then $X_{kl} = 1$ and constraints 24 guarantee that if $\sum_{t \in H} N_{klt} = 0$, then $X_{kl} = 0$.

$$X_{kl} + X_{k'l} \leq 1 \quad \forall (k, k') \notin S, l \in L \quad (22)$$

$$\sum_{t \in H} N_{klt} \leq X_{kl} M \quad \forall k \in K, l \in L_k \quad (23)$$

$$\sum_{t \in H} N_{klt} \geq X_{kl} \quad \forall k \in K, l \in L_k \quad (24)$$

Alternative Task-Unit Assignment (M1 and M2).

Moreover, we may want to ensure that from the alternative units available for each task only one is assigned. Constraints 25 guarantee that from the alternative tasks A_k to k only one is selected.

$$X_{kl} + \sum_{k' \in A_k} X_{k'l} \leq 1 \quad \forall k \in K, l \in L_k \quad (25)$$

7. NUMERICAL RESULTS

In order to show how general model M2 is and to compare its effectiveness, we consider four different chemical processes. Process 1 was first addressed by Kondili et al.,²⁵ Process 2 was published by Kallrath,³⁷ Process 3 was proposed in the paper of Papageorgiou and Pantelides,³⁸ and finally Process 4, depicted in Figure 1, is proposed by us. The first three processes are benchmark problems from the literature and fairly represent the existing scheduling complexities of the multipurpose batch plants. The last process is intended to allow an analysis of lots blending and traceability features and the model extensions.

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

Model M1 is defined by constraints A3 to A12, if changeovers are not present, and by constraints A1, A2, and A4 to A12, if changeovers are modeled. Model M2 is defined by constraints 1 to 11, 13, and 14, if changeovers are not required, and by constraints 1 to 10 and 12 to 14, if changeovers are needed. The objective function is to maximize the economical result of the global operation and is the same for both models, despite the modeling differences in the storage costs discussed in section 5.4.

The models were implemented using ILOG/CPLEX version 12.5, running on an Intel Xeon X5680 at 3.33 GHz with 24 GB of RAM. We have considered the time limit of 3600 s and the integrality gap of 5% as stopping criteria, so as to evaluate the models performance respecting the time to obtain solutions and their quality. The networks of processes P1, P2, and P3 and respective data tables are given in the Supporting Information.

7.1. Process 1. Process 1 is the network published by Kondili et al.²⁵ This process involves a cyclic material flow, alternative processing units, and different storage policies. Additionally, we have performed a slight modification of the network by considering the NIS policy for the intermediaries HOTA, INTBC, and IMPE. Because Process 1 has a unique network, no changeovers were defined. Moreover, we assume a single lot, thus lots blending are not considered and materials traceability is implicitly ensured.

Numerical results for Process 1 are depicted in Table 1 for the case where the stopping criterion is the time limit equal to 3600 s and in Table 2 where the stopping criterion is the integrality gap of 5%. As expected, model M2 always has less binary variables and more continuous variables and constraints when compared with model M1. This is because M1 makes use of binary variables to model storage tasks, while M2 implements storage through the set of constraints 11.

Table 1. Process 1 Solution Statistics (Stopping Criterion Is the Time Limit of 3600 s)

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

Table 2. Process 1 Solution Statistics (Stopping Criterion Is the Integrality Gap of 5%)

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

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

In the 120 and 240 h instances none of the models succeeded to prove optimality. In the horizon of 120 h, M1 is slightly better than M2, and in the 240 h instance, the solution of M2 is better than M1. Assuming a margin of 5% for the integrality gap, the 120 h instance of M1 reached a solution in 201 s, while M2 took 381 s. However, in the 240 h instance, M2 reached a better solution in just 460 s, while M1 required 2053 s.

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

7.2. Process 2. Process 2 was published by Kallrath³⁷ and is being extensively used as a benchmark problem because of its complexity. The process suggested by the author accounts for flexible output proportions for intermediaries, several storage policies, a cyclic material flow, and a considerable number of states, units, and tasks. In this paper, we do not consider flexible output proportions; therefore, the proportion of material going to State3 was fixed to 0.3 and the proportion of material going to State4 was fixed to 0.7. Again, we assume a single lot and that there are no sequence-dependent changeovers.

The solution statistics presented in Table 3 show that model M2 performed better than model M1 in all instances. In the 48 h horizon, M1 proved optimality in 233 s, while M2 just took 101 s. With the increase of the model size, both models had difficulties in reaching an optimal solution; however, M2 obtained always the best solution within the specified time limit. In the 120 h horizon, the solution obtained by M1 was within a gap of 12.86%, while the solution retrieved by M2 ensured a gap of 3.11%.

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

7.3. Process 3. Process 3 is from Papageorgiou and Pantelides³⁸ and is defined by three parallel production lines that share almost all processing units. The processes have several storage policies, including ZW and NIS, and have tasks with small and large processing times. Here, we consider sequence-

dependent changeovers between products that have a single lot, and we test these processes with time grids of 1 and 5 h.

In Table 4, we show the results with a time grid of 5 h and for scheduling horizons of 120 and 240 h. It is not possible to run Process 3 for smaller time horizons, because tasks have large processing times. Model M2 outperformed model M1 in both instances. In the 120 h horizon, M1 proved optimality in 5 s, which required 18 s. And in the 240 h horizon, M2 proved optimality in just 476 s, while M1 needed 677 s. The number of nodes and iterations of the branch-and-bound for model M2 are also significantly smaller when compared with those of model M1. Considering the stopping criterion of 5% in the integrality gap, Table 5, model M1 obtained a solution in just 10 s, while M2 required 58 s.

By assuming a time grid of 1 h, the model size naturally increased in a significant way and none of the models proved optimality; see Tables 6 and 7. M2 performed better than M1 in all instances, always reaching an integrality gap within 5%, with the exception of one instance.

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

7.4. Process 4. We now consider the network defined by the three processes depicted in Figure 1. Products PA and PB are produced from raw materials, while Product PC is produced from PA and PB. Moreover, we are given the unit's physical layout shown in Figure 6. This process is used to test the performance of both models and also to address new modeling features only possible to be treated with model M2 with the extensions proposed in Section 6.

First, we test Process 4 assuming sequence-dependent changeovers and single lots without blending. Since model M1 cannot address lots blending and traceability features, we slightly change the recipe of product PC by imposing that the materials required to produce PC are raw materials and not the products PA and PB as defined in Figure 1. The numerical results for this scenario are presented in Tables 8 and 9. Second, we define multiple lots and assume that lots blending may happen. Thus, here only model M2 is tested. We analyze lots traceability, sequence-dependent changeovers, temporary storage in the processing units, task-unit-layout, and alternative task-unit assignments. The numerical results for this case are shown in Table 12.

Single Lots without Blending. As it can be seen in Tables 8 and 9, the results obtained by model M2 are superior to the results retrieved by model M1. For example, in the 48 h horizon instance, the solution time of M2 is 651 s, while M1 required 1996 s.

In the 120 and 240 h horizons, none of models could prove optimality for the CPU time limit of 3600 s. Nevertheless, the solutions obtained by M2 are always better, achieving integrality

Table 3. Process 2 Solution Statistics (Stopping Criterion Is the Time Limit of 3600 s)

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

Table 4. Process 3 Solution Statistics (Time Grid Is 5 h)

model/process/horizon/grid	int. variables/cont. variables/constraints	nodes	iterations	LP relaxation	gap (%)	objective	CPU time (s)
M1/P3/120/5	1550/1720/2841	17,969	1,413,603	5,387.3	0.00	5,066.0	18
M2/P3/120/5	575/2879/5086	5,893	297,682	5,392.2	0.00	5,066.0	5
M1/P3/240/5	3038/3355/5541	330,548	40,797,510	10,774.7	0.00	10,184.8	677
M2/P3/240/5	1127/5642/9946	199,201	21,796,427	10,784.5	0.00	10,184.8	476

Table 5. Process 3 Solution Statistics (Time Grid Is 5 h and Stopping Criterion Is the Integrality Gap of 5%)

model/process/horizon/grid	nodes	iterations	gap (%)	objective	CPU time (s)
M1/P3/240/5	18,539	2,565,141	4.30	10,000.6	58
M2/P3/240/5	4,771	446,278	5.00	9,966.8	10

gaps that are less than 5% and that are less than half of the gaps obtained by M1.

With an integrality gap of 5% as stopping criterion, model M2 also performed better than M1, as it can be seen in Table 9. The solution times of M2 are considerably smaller than the solution times required by M1 (except for the 24 h instance). In the 120 h horizon, M1 required 2554 s and M2 required 51 s, and in the 240 h instance M1 needed 10 370 s, while M2 just needed 267 s.

The performance of the extensions, on changeovers start and nonpreemptive lots, expressed by constraints 16 and 17, respectively, is assessed by the numerical results of Table 10. Constraints 17 impose that lots cannot be interrupted to produce other lots, thus limiting the profit of the schedule when compared with the profit values shown in Table 8. The computational performance of the model M1.1 tends to decrease with the increase of the time horizon, as can be seen by the large integrality gaps of the 120 and 240 h instances.

On the Changeover Costs. In order to reflect the changeover costs on the schedule solutions, we have added expression 26 in the objective function of model M1. The cost structures of the resultant model M1.2 and of model M2 are illustrated in Figure 7.

$$GC = \sum_{r \in E} \sum_{l \in L_r} \sum_{l' \in L_r} \sum_{t \in H} (c_{rl'l't}) \quad (26)$$

It can be seen that in the 24 h instance both models had storage costs equal to 3032 m.u. In the 48 h storage costs increased to 6463 in model M2, while model M1.2 storage costs (SC) increased to 6807. Regarding the operational costs (OP), M2 had always inferior costs than M1.2. It is important to note that, when changeover costs (GC) are considered in the objective function the trade-offs between task-unit allocation, storage, and changeover costs pass to exist.

The computational results of model M1.2 are shown in Table 11. As expected, the profit obtained by M1.2 is always inferior to the profit obtained by models M1 and M2 (see Table 8) due to the changeover costs. Model M1.2 demonstrated worse performance than M1, particularly in the larger instances. For example, in the 240 h instance, M1.2 had 28.23% of integrality gap, in contrast to M1 that had 8.96%.

Looking into the scheduling solutions (see Figure 8), we can analyze how changeover costs affect the task-unit assignment. The schedule solution of M1.2 has a total of 3 changeovers, resulting into a cost of 800 m.u. and an idle time of 11 h. Although M2 does not model changeover tasks, costs and time of the changeovers can be derived by analyzing the schedule solution. In this way, the schedule solution of M2 has a total of 7 changeovers that result into a cost of 1800 m.u. and an idle time of 25 h. Processing units are used less efficiently in M2, which concerns to the total changeover time and costs. Nevertheless, the profit of M2 is 99.8% of M1.2, discounting the changeover cost of 1800 m.u. to the profit obtained by M2. Thus, although M1.2 and M2 schedules are slightly different, they deliver the same amount of products and have a similar profit. In practice, since changeover constraints lead to a more efficient model, they can be used instead of changeover tasks if (a) the exact time of the changeover is not relevant; (b) utilities/materials consumption during changeovers can be disregarded; and (c) changeover costs are not significant.

Multiple Lots with Blending. Now, we use the processes as shown in Figure 1 to obtain schedules with multiple lots per product and with blending operations. We considered the production of two lots of PA, two lots of PB, and a single lot of PC, in a time horizon of 48 h. The aim is to define production schedules in which the traceability of lots is kept during the entire horizon and the tasks-units assignment is done by assuming the physical layout limitations shown in Figure 6. For that we consider the extended model M2.1 by adding constraints 22 to 25 to model M2. Moreover, we also include the lot sizes extensions in model M2.2 that are defined by constraints 19 to 25.

Figure 9 shows the schedule for a 48 h horizon, having an objective value of 340 442.2 m.u., relative to a delivery of 7000 kg of product PC. Lot L1 of product PA starts first in units U2 and F1, while lot L2 of the same product is processed in units U4 and F2. We can see that the physical layout limitations expressed in Figure 6 were followed by both lots. Regarding the production of PB, lots L1 and L2 were produced in units U3, U3, and F2.

Although model M2.1 does not explicitly give the start of the temporary storage tasks in the processing units and the sequence-dependent changeovers, those can be directly deduced from R_{krlt} and N_{klt} variables. Thus, it can be seen that intermediary PA_S3 is temporarily stored in unit U3 in all occurrences of TASK3 of product PA. Concerning the sequence-dependent changeovers, we can see changeovers between lots of different products and changeovers between lots of the same product. This latter case happens in unit U3 at the time interval 9.

Table 6. Process 3 Solution Statistics (Time Grid Is 1 h and Stopping Criterion Is the Time Limit of 3600 s)

model/process/horizon/grid	int. variables/cont. variables/constraints	nodes	iterations	LP relaxation	gap (%)	objective	CPU time (s)
M1/P3/120/1	7502/8248/13653	178,586	72,707,311	5,317.2	13.34	4,587.7	3,600
M2/P3/120/1	2783/13919/30346	168,092	25,231,890	5,321.2	9.09	4,768.5	3,600
M1/P3/240/1	14942/16411/27165	56,025	26,545,793	10,634.3	11.18	9,457.9	3,600
M2/P3/240/1	5543/27722/60418	59,941	17,039,145	10,642.4	5.80	9,920.5	3,600

Table 7. Process 3 Solution Statistics (Time Grid Is 1 h and Stopping Criterion Is the Integrality Gap of 5%)

model/process/horizon/grid	nodes	iterations	gap (%)	objective	CPU time (s)
M1/P3/120/1	578,271	296,170,781	7.84	4,795.6 ^a	14,400
M2/P3/120/1	566,936	131,212,184	4.78	4,912.4	14,400
M1/P3/240/1	341,515	136,133,027	10.24	9,496.7 ^a	14,400
M2/P3/240/1	129,924	44,592,826	4.98	9,978.7	7,125

^aStopping criterion is the time limit of 14,400 s.

Table 8. Process 4 Solution Statistics (Stopping Criterion Is the Time Limit of 3600 s)

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

Table 9. Process 4 Solution Statistics (Stopping Criterion Is the Integrality Gap of 5%)

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

Table 10. Process 4 Solution Statistics, Assuming Changeovers Start and Non-Preemptive Lots Constraints

model/process/horizon/grid	constraints	gap (%)	objective	CPU time (s)
M1.1/P4/24/1	3,545	0.00	501,290.0	10
M1.1/P4/48/1	6,914	0.57 ^a	998,965.5	3,600
M1.1/P4/120/1	17,099	23.02 ^a	2,060,248.0	3,600
M1.1/P4/240/1	34,100	34.94 ^a	3,692,146.0	3,600

^aStopping criterion is the time limit of 3,600 s.

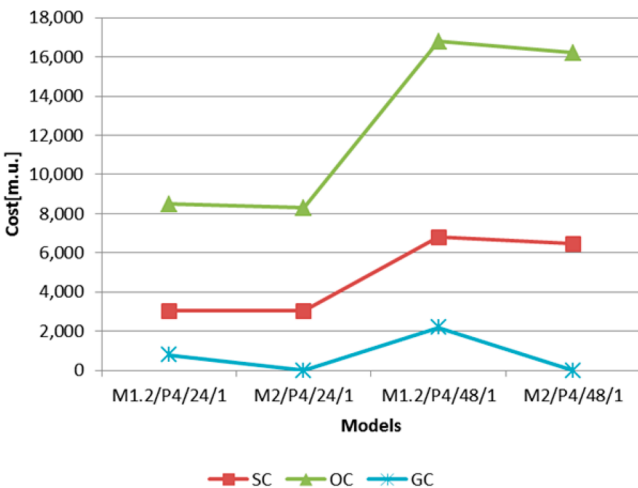


Figure 7. Cost structure for models M1.2 and M2 (SC, storage costs; OC, operational costs; GC, changeover costs).

Finally, lots traceability is ensured for all products. The amounts produced of PA and PB of each lot are consumed by product PC and are directly traceable. For example, at the time interval 27, the amount of lot L1 of product PB is 2500 kg and of L2 is 833.3 kg, and because the amount of L1 of PB is not sufficient to feed the batch of TASK2 of PC, it is necessary to blend lots. This situation can be seen in Figure 10, at time 28, where lots L1 and L2 of PB are consumed simultaneously by TASK2 of product PC.

Table 12 shows the computational results for models M2.1 and M2.2. M2.1 obtained a profit equal to 340 442.2 after 3,600 s. But assuming an integrality gap of 5%, a solution was retrieved in just 31 s. Model M2.2 takes into account the lot sizes constraints 19 and 20, and it can be seen that M2.2 and M2.1 performances are comparable for the tested instance.

8. CONCLUSIONS

In this work, we propose two general discrete-time scheduling models for multipurpose batch plants (models M1 and M2). We first use a RTN discrete-time formulation (M1) as basis for comparison with a more innovative model (M2). The first model (Model M1) extends the RTN model of Pantelides,²⁷ by considering explicitly and in an integrated way scheduling features already treated in the literature, such as temporary storage in the processing units,²⁵ sequence-dependent changeovers,³⁰ and multiproduct delivery extensions.³¹ This model is then generalized by considering the start of changeovers tasks and nonpreemptive lots, as well as alternative task-unit and task-unit-layout assignments.

Table 11. Process 4 Solution Statistics, Assuming Changeovers Costs

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

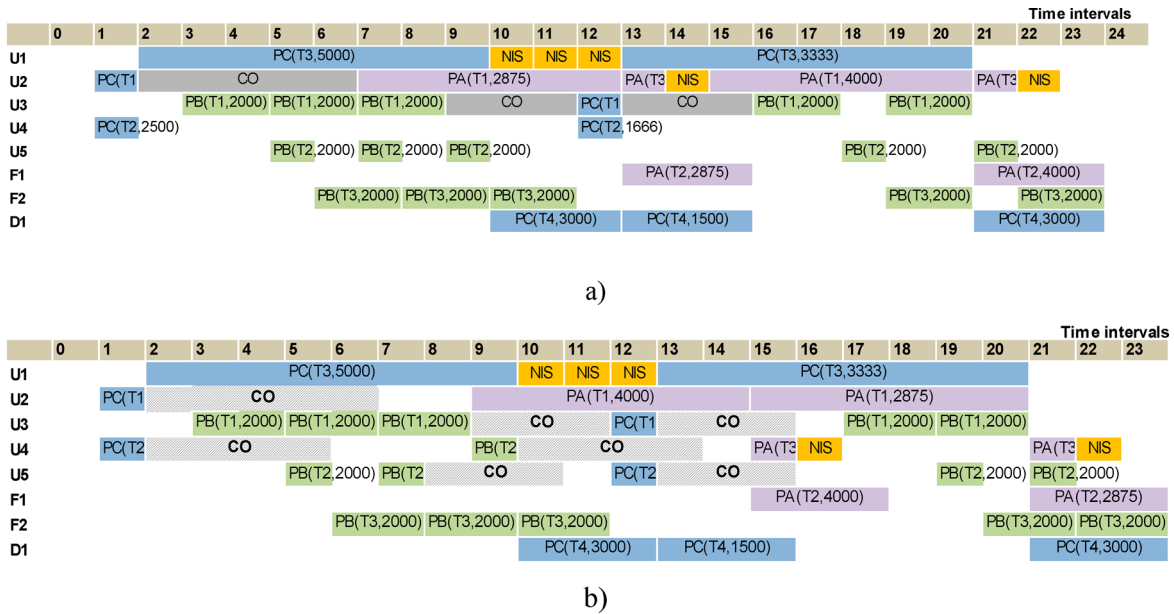


Figure 8. Scheduling for 24 h instance: (a) model M1.2 and (b) model M2 (CO = changeover).

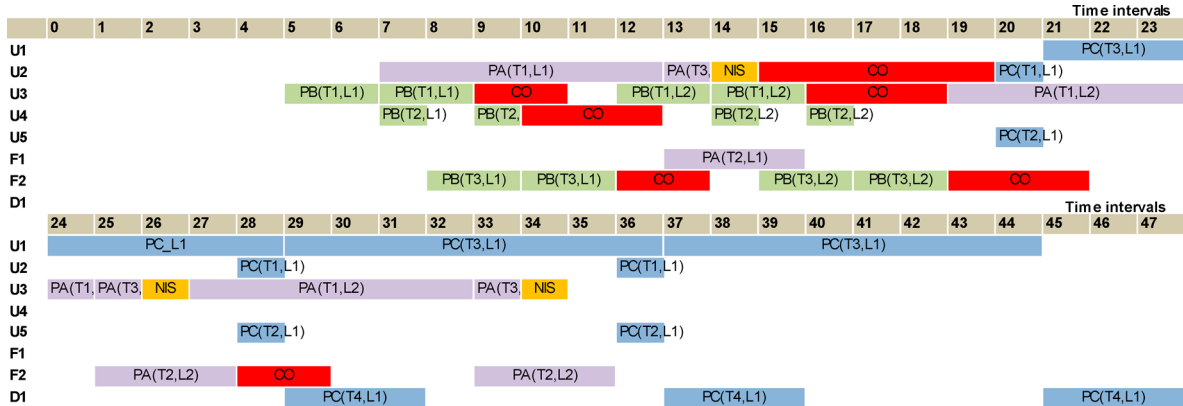


Figure 9. Scheduling for instance M2.1/P4/48/1^a).

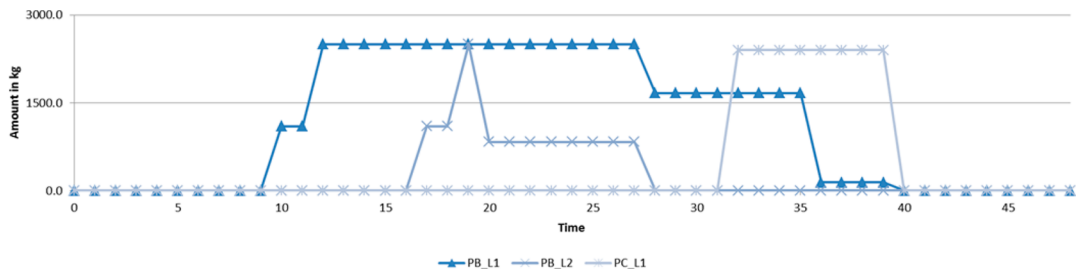


Figure 10. Inventory for lots of products PB and PC.

Model M2, based on STN, can be viewed as an innovative contribution in the area, explicitly modeling the inventory carried out in each task by adding a task index to the resource availability variables. This approach allows the development of new types of

constraints for modeling sequence-dependent changeovers and temporary storage in the processing units. Moreover, we address lots blending, lots start, and alternative task-unit and task-unit-layout assignments. Lots blending and traceability are two

Table 12. Process 4 Solution Statistics

model/process/horizon/grid	int. variables/cont. variables/constraints	nodes	iterations	gap (%)	objective	CPU time (s)
M2.1/P4/48/1 ^a	1650/8875/30554	36,169	31,493,851	0.61	340,442.2	3,600
M2.1/P4/48/1 ^b	1650/8875/30554	935	233,990	2.36	338,356.4	31
M2.2/P4/48/1 ^a	1650/8875/30580	57,805	34,439,408	0.14	338,787.2	3,600
M2.2/P4/48/1 ^b	1650/8875/30580	236	181,674	4.13	332,480.2	34

^aStopping criterion is the time limit of 3,600 s. ^bStopping criterion is the integrality gap of 5%.

requirements introduced in this work that are common in the chemical and biochemical–pharmaceutical industries, considered here with the purpose of keeping record of the blending processes during the production.

We compare the effectiveness of both models using three benchmark problems from the literature and one scheduling problem proposed in this paper. Experimental results have shown that model M2 is computationally more effective for the instances tested. In the larger or more complicated instances, both models had difficulties in proving optimality. However, model M2 always reached a solution within 5% of the integrality gap, except for the 240 h scheduling horizon of Kallrath³⁷ network. Model M1 had worse performance in most of the cases.

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

APPENDIX: RTN MODEL (M1)

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

$$\sum_{l \in L_r} R_{rl}^{\text{init}} \leq 1 \quad \forall r \in E \quad (\text{A2})$$

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

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

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

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

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

$$Q_{rld}^{\text{min}} - \Pi_{rld}^{\text{slack}} \leq \sum_{t \in DW_{rld}} (-\Pi_{rlt}) \leq Q_{rld}^{\text{max}} \quad \forall r \in P, l \in L_r, d \in D_r$$

$$DW_{rld} = \{t | r \in P, l \in L_r, d \in D_r, t \in H: T_{rd}^{dd} \geq t \geq T_{rd}^{\text{ed}}\} \quad (\text{A8})$$

$$\Pi_{rlt} = 0 \quad \forall r \in P, l \in L_r, d \in D_r, t \in H \setminus DW_{rld} \quad (\text{A9})$$

$$\Pi_{rlt} = 0 \quad \forall r \in I, l \in L_r, t \in H \quad (\text{A10})$$

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

$$R_{rlt} \in \mathbb{R}_+ \quad \forall r \in E \cup I \cup P, l \in L_r, t \in H$$

$$\xi_{klt} \in \mathbb{R}_+ \quad \forall r \in E, k \in K_r, l \in L_k, t \in H$$

$$\Pi_{rld}^{\text{slack}} \in \mathbb{R}_+ \quad \forall r \in P, l \in L_r, d \in D_r$$

$$\Pi_{rlt} \in \mathbb{R}_- \quad \forall r \in I \cup P, l \in L_r, t \in H$$

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

$$C_{rl'l''t} \in \{0, 1\} \quad \forall r \in E, l', l'' \in L_r, t \in H \quad (\text{A12})$$

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

The materials balance constraints A4 are similar to the units balance constraints A1 or A3. The difference is that constraints A4 handle intermediaries and final products and not processing units. Materials are consumed and produced at the proportion $\nu_{kr\theta}$ of the batch size ξ_{klt} . The continuous variables Π_{rlt} express the deliveries of product r of lot l at the time interval t and will always have non-positive values; thus, no material receipts are expected to occur during the scheduling horizon. We opted not to model

raw materials since it can be assumed, without loss of generality, that raw materials are available when needed. Constraints A5 define the minimum and maximum materials availability allowed for each time interval. These constraints also permit the definition of different storage policies depending on the value of parameters R_{rt}^{\max} . Thus, R_{rt}^{\max} take the value 0 for Non-Intermediate Storage (NIS) or Zero-wait (ZW) and takes a value greater than zero if there is Finite Intermediate Storage (FIS) or Unlimited Intermediate Storage (UIS). In the latter case the value should be sufficiently large to account for unlimited storage capacity. Constraints A6 define that the batch size ξ_{kl} must be within the minimum V_{krl}^{\min} and maximum V_{krl}^{\max} allowed capacities of resource r and task k of lot l .

Constraints A7 were first proposed by Kondili et al.²⁵ to model temporary storage done by the processing units (NIS policy) and ensure that the intermediary is held by the unit in which it was produced. These constraints require the creation of additional storage tasks to model the NIS policy and impose that the batch size of a storage task is less than or equal to the previous amount stored plus the amount produced at each time interval. If the batch size of a storage task is greater than zero, then the assignment binary variable for the storage task must be one by constraints A6. Parameters $\nu_{kr\theta}^p$ give the production proportion of the batch size of task k for resource r , and I^{NIS} is a subset of I that has the intermediaries subject to the NIS policy. Note that storage tasks have duration equal to one since materials availability needs to be checked at every time interval. These constraints are only required in the cases that the alternative units suitable to perform a given task are dissimilar. In these situations, constraints A7 guarantee that the unit allocated during the storage period is the same unit that has produced the material being held.

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

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

■ ASSOCIATED CONTENT

■ Supporting Information

Networks of processes P1, P2, and P3 and data tables. This material is available free of charge via the Internet at <http://pubs.acs.org>.

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: apovoa@tecnico.ulisboa.pt.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The authors gratefully acknowledge the financial support of Hovione FarmaCiencia SA, Fundação para a Ciência e Tecnologia, under Grant SFRH/BD/33970/2009, and the North Portugal Regional Operational Programme (ON.2 – O

Novo Norte), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF).

■ NOTATION

Indices

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

Sets

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

Parameters

$\alpha_{rl'l''\theta}$ allocation/release changeover coefficient of resource r (processing unit) from lot l' to l'' being at lot l and at time θ relative to the start of the changeover task
 $\mu_{kr\theta}$ allocation/release coefficient of resource r (processing unit) in task k at time θ relative to the start of the task
 τ_k processing time of task k
 $\nu_{kr\theta}^p, \nu_{kr\theta}^c, \nu_{kr\theta}^s$ production/consumption proportion of resource (intermediary or final product) r in task k at time θ relative to the start of task
 C_r^{sto} cost of storage of products and intermediaries r
 C_k^{op} operational costs of task k
 C_{rld}^{slack} missing deliveries cost for material r of lot l and delivery d
 $C_{rl'l'}$ changeover time in processing unit r from lot l to lot l'
 $Q_{rld}^{\min}, Q_{rld}^{\max}$ minimum and maximum amount of lot l and product r at delivery d

R_{rt}^{\max}	maximum resource availability of resource r (intermediary or final product) at time interval t
$R_{rl}^{\text{init}(m)}$	resource r (intermediary or final product) availability of lot l in the beginning of the planning horizon
$R_{rl}^{\text{init}(m)}$	resource r (intermediary or final product) availability of lot l at task k in the beginning of the planning horizon
T	length of the scheduling horizon
T_{ld}^{ed}	earliest time interval of lot l at delivery d
T_{ld}^{ld}	latest time interval of lot l at delivery d
ν_r	value of product r
$V_{krl}^{\min}, V_{krl}^{\max}$	minimum and maximum capacity of resource r (processing unit) for task k of lot l

Variables

ξ_{klt}	batch size of task k and lot l at time interval t (continuous) (models M1 and M2)
Π_{rlt}	delivery of resource (final products) r of lot l at time interval t (continuous) (model M1)
Π_{krlt}	delivery of resource (final products) r of lot l at time interval t available from task k (continuous) (model M2)
Π_{rlt}^{slack}	missing delivery d of lot l of product r (continuous) (models M1 and M2)
$C_{rll't}$	binary variables that are equal to 1 if a changeover task occur on resource (processing units) r between lots l and l' (model M1)
N_{klt}	binary variables that are equal to 1 if task k starts lot l at time interval t (models M1 and M2)
R_{rl}^{init}	allocation of resource r (processing unit) at the beginning of the scheduling horizon (continuous) (model M1)
R_{rlt}	resource availability r at lot l and at time interval t (continuous) (model M1)
R_{rt}	resource availability (processing units) r at time interval t (continuous) (model M1)
R_{krlt}	resource r (intermediaries or final products) availability, produced by task k of lot l at time interval t (continuous) (model M2)
R_{krlt}^c	amount of resource r (intermediaries or final products) consumed from task k of lot l at time interval t (continuous) (model M2)
R_{krlt}^p	amount of resource r (intermediaries or final products) produced by task k of lot l at time interval t (continuous) (model M2)
X_{kl}	binary variables that are equal to 1 if task k is assigned to lot l (models M1 and M2)

REFERENCES

- (1) Grossmann, I. Advances in mathematical programming models for enterprise-wide optimization. *Comput. Chem. Eng.* **2012**, *47*, 2–18.
- (2) Varma, V.; Reklaitis, G.; Blau, G.; Pekny, J. Enterprise-wide modeling & optimization—An overview of emerging research challenges and opportunities. *Comput. Chem. Eng.* **2007**, *31* (5–6), 692–711.
- (3) Mendez, C. A.; Cerda, J.; Grossmann, I. E.; Harjunkski, I.; Fahl, M. State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Comput. Chem. Eng.* **2006**, *30* (6–7), 913–946.
- (4) Barbosa-Povoa, A. P. A critical review on the design and retrofit of batch plants. *Comput. Chem. Eng.* **2007**, *31* (7), 833–855.
- (5) Li, Z.; Ierapetritou, M. Process scheduling under uncertainty: Review and challenges. *Comput. Chem. Eng.* **2008**, *32* (4), 715–727.
- (6) Maravelias, C. T.; Sung, C. Integration of production planning and scheduling: Overview, challenges and opportunities. *Comput. Chem. Eng.* **2009**, *33* (12), 1919–1930.

- (7) Verderame, P. M.; Elia, J. A.; Li, J.; Floudas, C. A. Planning and Scheduling under Uncertainty: A Review Across Multiple Sectors. *Ind. Eng. Chem. Res.* **2010**, *49* (9), 3993–4017.
- (8) Grossmann, I. E. Review of nonlinear mixed-integer and disjunctive programming techniques. *Optim. Eng.* **2002**, *3* (3), 227–252.
- (9) Méndez, C.; Henning, G.; Cerda, J. An MILP continuous-time approach to short-term scheduling of resource-constrained multistage flowshop batch facilities. *Comput. Chem. Eng.* **2001**, *25* (4–6), 701–711.
- (10) Méndez, C. A.; Cerda, J. Dynamic scheduling in multiproduct batch plants. *Comput. Chem. Eng.* **2003**, *27* (8–9), 1247–1259.
- (11) Castro, P.; Erdirlik-Dogan, M.; Grossmann, I. Simultaneous batching and scheduling of single stage batch plants with parallel units. *AIChE J.* **2008**, *54* (1), 183–193.
- (12) Sundaramoorthy, A.; Maravelias, C. T. Simultaneous batching and scheduling in multistage multiproduct processes. *Ind. Eng. Chem. Res.* **2008**, *47* (5), 1546–1555.
- (13) Pinto, J. M.; Grossmann, I. E. A continuous time mixed integer linear programming model for short term scheduling of multistage batch plants. *Ind. Eng. Chem. Res.* **1995**, *34* (9), 3037–3051.
- (14) Liu, Y.; Karimi, I. Novel continuous-time formulations for scheduling multi-stage batch plants with identical parallel units. *Comput. Chem. Eng.* **2007**, *31* (12), 1671–1693.
- (15) Liu, Y.; Karimi, I. Scheduling multistage batch plants with parallel units and no interstage storage. *Comput. Chem. Eng.* **2008**, *32* (4), 671–693.
- (16) Ierapetritou, M.; Floudas, C. Effective continuous-time formulation for short-term scheduling. 1. Multipurpose batch processes. *Ind. Eng. Chem. Res.* **1998**, *37* (11), 4341–4359.
- (17) Janak, S. L.; Lin, X.; Floudas, C. A. Enhanced continuous-time unit-specific event-based formulation for short-term scheduling of multipurpose batch processes: Resource constraints and mixed storage policies. *Ind. Eng. Chem. Res.* **2004**, *43* (10), 2516–2533.
- (18) Shaik, M. A.; Floudas, C. A. Improved unit-specific event-based continuous-time model for short-term scheduling of continuous processes: Rigorous treatment of storage requirements. *Ind. Eng. Chem. Res.* **2007**, *46* (6), 1764–1779.
- (19) Vooradi, R.; Shaik, M. A. Improved Three-Index Unit-Specific Event-Based Model for Short-Term Scheduling of Batch Plants. *Comput. Chem. Eng.* **2012**, *43*, 148–172.
- (20) Castro, P.; Barbosa-Póvoa, A. P.; Matos, H. A.; Novais, A. Q. Simple continuous-time formulation for short-term scheduling of batch and continuous processes. *Ind. Eng. Chem. Res.* **2004**, *43* (1), 105–118.
- (21) Schilling, G.; Pantelides, C. A simple continuous-time process scheduling formulation and a novel solution algorithm. *Comput. Chem. Eng.* **1996**, *20*, S1221–S1226.
- (22) Castro, P. M.; Harjunkski, I.; Grossmann, I. E. New continuous-time scheduling formulation for continuous plants under variable electricity cost. *Ind. Eng. Chem. Res.* **2009**, *48* (14), 6701–6714.
- (23) Maravelias, C. T.; Grossmann, I. E. New general continuous-time state-task network formulation for short-term scheduling of multipurpose batch plants. *Ind. Eng. Chem. Res.* **2003**, *42* (13), 3056–3074.
- (24) Sundaramoorthy, A.; Karimi, I. A simpler better slot-based continuous-time formulation for short-term scheduling in multipurpose batch plants. *Chem. Eng. Sci.* **2005**, *60* (10), 2679–2702.
- (25) Kondili, E.; Pantelides, C.; Sargent, R. A general algorithm for short-term scheduling of batch operations—I. MILP formulation. *Comput. Chem. Eng.* **1993**, *17* (2), 211–227.
- (26) Shah, N.; Pantelides, C.; Sargent, R. A general algorithm for short-term scheduling of batch operations—II. Computational issues. *Comput. Chem. Eng.* **1993**, *17* (2), 229–244.
- (27) Pantelides, C. C. In *Unified frameworks for optimal process planning and scheduling*; Cache Publications: New York, 1994; pp 253–274.
- (28) Barbosa-Povoa, A. P.; Macchietto, S. Detailed design of multipurpose batch plants. *Comput. Chem. Eng.* **1994**, *18* (11–12), 1013–1042.
- (29) Pinto, T.; Barbosa-Póvoa, A. P. F. D.; Novais, A. Q. Optimal design and retrofit of batch plants with a periodic mode of operation. *Comput. Chem. Eng.* **2005**, *29* (6), 1293–1303.

- (30) Castro, P.; Novais, Q.; Carvalho, A. Optimal equipment allocation for high plant flexibility: An industrial case study. *Ind. Eng. Chem. Res.* **2008**, *47* (8), 2742–2761.
- (31) Wassick, J. M.; Ferrio, J. Extending the Resource Task Network for Industrial Applications. *Comput. Chem. Eng.* **2011**, *35* (10), 2124–2140.
- (32) Sundaramoorthy, A.; Maravelias, C. T. A general framework for process scheduling. *AIChE J.* **2011**, *57* (3), 695–710.
- (33) Moniz, S.; Barbosa-Póvoa, A. P.; Pinho de Sousa, J. Regular and non-regular production scheduling of multipurpose batch plants. *Proceedings of the 22nd European Symposium on Computer Aided Process Engineering*; IChemE: 2012.
- (34) Floudas, C. A.; Lin, X. Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review. *Comput. Chem. Eng.* **2004**, *28* (11), 2109–2129.
- (35) Sundaramoorthy, A.; Maravelias, C. T. Computational Study of Network-Based Mixed-Integer Programming Approaches for Chemical Production Scheduling. *Ind. Eng. Chem. Res.* **2011**, *50* (9), 5023–5040.
- (36) Castro, P.; Barbosa-Póvoa, A. P.; Matos, H. A. Optimal periodic scheduling of batch plants using RTN-based discrete and continuous-time formulations: A case study approach. *Ind. Eng. Chem. Res.* **2003**, *42* (14), 3346–3360.
- (37) Kallrath, J. Planning and scheduling in the process industry. *OR Spectrum* **2002**, *24* (3), 219–250.
- (38) Papageorgiou, L. G.; Pantelides, C. C. A hierarchical approach for campaign planning of multipurpose batch plants. *Comput. Chem. Eng.* **1993**, *17*, S27–S32.