Production Planning and Scheduling of Parallel Continuous Processes with Product Families

Georgios M. Kopanos,[†] Luis Puigjaner,[†] and Christos T. Maravelias*,[‡]

Department of Chemical Engineering, Universitat Politècnica de Catalunya, ETSEIB, Av. Diagonal 647, Barcelona 08028, Spain, and Department of Chemical & Biological Engineering, University of Wisconsin, Madison, Wisconsin 53705

The goal of this paper is the development of a mixed integer programming formulation for the production planning and scheduling of parallel (single-stage) continuous processes in the presence of sequence-dependent switchover times and costs for product families, and sequence-independent switchovers for products belonging to the same family. Our formulation combines a discrete-time planning (big-bucket) grid with a continuous-time treatment of the scheduling decisions within each period and across adjacent periods. At the production planning level, it handles product orders at intermediate due dates and accounts for holding and backlog costs. At the scheduling level, it accounts for equipment unit constraints, switchover times and costs, maintenance activities, and idle production periods. The proposed model was motivated from and implemented to a real-world production facility. It can effectively address industrial-scale planning–scheduling problems. Importantly, it yields solutions that are considerably better than the solutions currently obtained using commercial tools.

1. Introduction

The goal of this paper is the development of an approach to the simultaneous production planning and scheduling of continuous parallel units producing a large number of final products that can be classified into product families. The problem at hand appears in many stages of operation in process industries, including packaging in batch and continuous production facilities. Thus, it is quite important since it arises in a number of different production environments (e.g., food and beverage industry, consumer products, etc.).

The objective in production planning is to determine the production amounts and inventory levels that will allow us to fulfill given customer demand at the minimum cost (including processing, holding and backlog, and switchover costs) subject to (typically aggregate) production capacity constraints. Thus, a production planning solution consists of production targets and inventory levels over a number of periods into which the planning horizon under consideration is partitioned. On the other hand, the objective in scheduling is the allocation of limited resources (e.g., equipment units, utilities, manpower) to competing tasks and the sequencing and timing of tasks on units, given a set of production targets and subject to detailed production constraints. Clearly, the two problems are interdependent since the solution of production planning (production targets) is input to scheduling, and the production capacity constraints in production planning depend on the scheduling solution.

In the realm of the process systems engineering (PSE) literature, a number of researchers have proposed standalone scheduling as well as integrated production planning–scheduling approaches. Mendez et al.[1] provide an excellent overview of process scheduling approaches, while Shah,[2] Kallrath,[3] and Maravelias and Sung[4] provide overviews of production planning and scheduling approaches. The reader is also pointed to Pochet and Wolsey[5] for a thorough discussion of mixed integer programming (MIP) methods to production planning.

In this paper we focus on systems with parallel units, often also termed single-stage processes. Given the prominence of this class of problems, a number of approaches have been discussed in the PSE literature for similar problems, though no methods have been reported for the specific problem discussed in the present paper.

Chen et. al[6] studied the medium-term planning problem of a single-stage single-unit continuous multiproduct polymer plant. They proposed a slot-based MIP model based on a hybrid discrete/continuous time representation, where the production planning horizon was divided into several discrete weeks, and each week was formulated with a continuous time representation. Liu et al.[7] further improved the work of Chen et al.[6] by presenting a MIP formulation based on the classic traveling salesman problem (TSP) formulation. Their proposed model, without time slots, was computationally more effective. Recently, Liu et al.[8] extended that work for medium-term planning of a single-stage multiproduct continuous plant to the case with parallel units. Erdirik-Dogan and Grossmann[9] proposed a multiperiod slot-based MIP model for the simultaneous planning and scheduling of single-stage single-unit multiproduct continuous plants. A bilevel decomposition algorithm in which the original problem is decomposed into an upper level planning and a lower level scheduling problem was also developed in order to deal with complex problems. Erdirik-Dogan and Grossmann[10] later extended their work to address parallel units.

In this paper, we study a more general case by considering product families, short planning periods that may lead to idle units for entire periods and changeovers spanning multiple periods, and maintenance activities. The motivation to consider product families comes from the fact that, in many production environments, there exist products that share many characteristics. In fact, the current work was first developed to address problems in a highly complex real-life bottling facility producing hundreds of final products. The grouping into families can be based on various criteria, including product similarities, processing similarities, or changeover considerations. The goal of the aforementioned grouping is to lead to computationally tractable optimization models without compromising the quality of

* To whom correspondence should be addressed. E-mail: maravelias@wisc.edu.
† Universitat Politècnica de Catalunya.
‡ University of Wisconsin.

solution. Furthermore, the use of product families reflects managerial practice prevalent in many production systems.[11−14]

The remainder of this paper is structured as follows. In section 2, we formally define the production planning−scheduling problem addressed in this paper. In section 3, we discuss the main elements of the proposed approach, and in section 4 we present the resulting MIP formulation. Finally, an illustrative example and two instances of an industrial case study are discussed in section 5.

2. Problem Statement

We consider the production planning−scheduling of continuous parallel units. The problem is defined in terms of the following items:

(i) A known planning horizon which is divided into a set of periods, $n \in N$.

(ii) A set of parallel processing units, $j \in J$, is included, with available production time in period n equal to ω_{jn}.

(iii) A set of product families or simply *families*, $f \in F$, wherein all products are grouped into; $f \in F_j$ is the subset of families that can be assigned to unit j and $j \in J_f$ is the subset of units that can process family f.

(iv) A set of products $i \in I$ with demand ζ_{in} at the end of time period n, backlog ψ_{in} and inventory ξ_{in} costs, minimum and maximum production rates ($\rho_{ijn}^{\min}$, $\rho_{ijn}^{\max}$), minimum processing times $\tau_{ijn}^{\min}$, and processing cost λ_{ij}; the subset of products in family f is denoted by $i \in I_f$; $i \in I_j$ is the subset of products that can be assigned to unit j, and $j \in J_i$ is the subset of units that can produce product i.

(v) A sequence-dependent switchover operation, henceforth referred to as *changeover*, is required on each processing unit whenever the production is changed between two different product families; the required changeover time is $\gamma_{ff'j}$, while the changeover cost is $\phi_{ff'j}$.

(vi) A sequence-independent switchover operation, henceforth referred to as *setup*, is required whenever a product i is assigned to a processing unit j; the setup time is δ_{ij} and the setup cost is θ_{ij}.

We assume a nonpreemptive operation mode, no utility restrictions, and that changeover times are not greater than a planning period.

Our goal is to determine the following, to satisfy customer demand at the minimum total cost, including operating, changeover, and setup costs, as well as inventory and backlog costs: (i) assignment (Y_{fjn}^F) of product families on each unit in every production period; (ii) sequencing ($X_{ff'jn}$) between families on each unit in every period; (iii) assignment of products to processing units in every period (Y_{ijn}); (iv) production amount for every product in every period (P_{in}); (v) inventory (S_{in}) and backlog (B_{in}) profiles for all products.

3. Proposed Approach

The novelty of our formulation lies in the integration of three different modeling approaches. In particular, we use a discrete-time approach for inventory and backlog cost calculation for production planning, a continuous-time approach with sequencing using immediate precedence variables for the scheduling of families, and a lot-sizing type of capacity constraints for the scheduling of products (see Figure 1).

For the production planning subproblem we employ a time grid with fixed, though not necessarily equal, production periods. The planning horizon is divided into $n \in N = \{1, 2, ...\}$ periods; period n starts at time point $n-1$ and finishes at time point n.
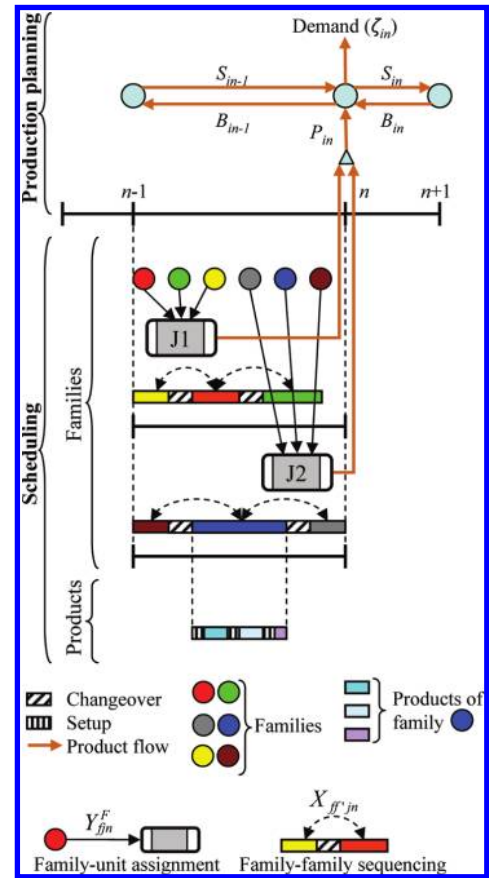

Figure 1. Proposed modeling approach.

Note that the use of a discrete-time approach at the planning level (big-bucket) enables the correct calculation of holding and backlog costs. Material balances for each product are expressed at the end of each planning period in terms of total production level, P_{in}, inventory level, S_{in}, and backlog level, B_{in}. We assume that the amounts produced during a planning period become available at the end of this period. The *communication* between the production planning and scheduling subproblems is accomplished via the amount Q_{ijn} of product i produced in unit j during period n. Variable Q_{ijn} is used by the production planning problem for the calculation of variables P_{in} for the material balances, while at the same time variables Q_{ijn} are subject to detailed sequencing and capacity constraints of the scheduling subproblem.

The scheduling subproblem has two levels. At the first level, we schedule product families on units using an immediate precedence approach for sequencing, while at the second level, we employ a lot-sizing-based approach to express capacity constraints for products. In particular, we define assignment binary variable Y_{fjn}^F to denote the assignment of family f in unit j during period n and sequencing binary variable $X_{ff'jn}$ ($\bar{X}_{f'fjn}$) to denote an immediate precedence $f \to f'$ in unit j within period n (across periods $n-1$ and n). This allows us to account for changeover times between families and correctly calculate changeover costs. The activation of the sequencing variables is achieved using a modification of the network formulation of Karmarkar and Schrage[15] and Sahinidis and Grossmann.[16] For the timing of processing of family f in unit j during period n, we introduce variable C_{fjn}. Individual product setups and capacity constraints are modeled using lot-sizing-type setup binary variables: $Y_{ijn} = 1$ if product i is produced in unit j during period n. Variables Y_{ijn} are used to constrain the processing time T_{ijn}, which is in turn used to constrain the production amount Q_{ijn}.

In addition to this novel integration of modeling approaches, our formulation can accurately account for changeovers in the presence of fixed planning periods. First, it allows us to track the last family produced in each period, and therefore account correctly for the first changeover in the following period. If the last family in period $n-1$ and the first in period n are the same, then no changeover time/cost is added (changeover carryover). Second, it allows changeover operations to cross planning period boundaries (changeover crossover), thereby allowing higher utilization of resources and obtaining better solutions (see Sung and Maravelias[17] for a discussion of these aspects in the context of lot-sizing problems). Finally, an extension of our approach allows the seamless modeling of idle planning periods, another aspect that has often been neglected, though it can appear in optimal solutions when the length of the planning period is short and/or a unit is not heavily loaded.

4. Mathematical Formulation

In this section, we present a MIP formulation for the production planning and scheduling of the parallel-unit (single-stage) facility described in section 2. Constraints are grouped according to the type of decision (e.g., assignment, timing, sequencing, etc.). To facilitate the presentation of the model, we use uppercase Latin letters for optimization variables and sets, lowercase Latin letters for indices, and lowercase Greek letters for parameters.

4.1. Material Balance Constraints. The total amount, P_{in}, of product i produced in period n is the summation of the produced quantities, Q_{ijn}:

$$P_{in} = \sum_{j \in J_i} Q_{ijn} \qquad \forall i, n \qquad (1)$$

Mass balances for every product i are expressed at the end of each production period n, where initial backlogs, $B_{in=0}$, and initial inventories, $S_{in=0}$, are given:

$$S_{in} - B_{in} = S_{in-1} - B_{in-1} + P_{in} - \zeta_{in} \qquad \forall i, n \qquad (2)$$

Inventory capacity constraints can be enforced via constraints similar to

$$S_{in} \leq \text{product storage capacity} \quad \forall i, n \quad \text{or}$$
$$\sum_i S_{in} \leq \text{plant storage capacity} \quad \forall n \qquad (3)$$

4.2. Family Allocation Constraints. Obviously, a family f is assigned to a processing unit $j \in J_f$ during a production period n if at least one product that belongs to this family, $i \in I_f$, is produced on this processing unit at the same period:

$$Y^F_{fjn} \geq Y_{ijn} \qquad \forall f, i \in I_f, j \in J_f, n \qquad (4)$$

where Y^F_{fjn} denotes a family to-unit assignment, and Y_{ijn} denotes a product-to-unit assignment.

Moreover, constraint 5 enforces the binary Y^F_{fjn} to be zero when no product $i \in I_f$ is produced on unit j in period n:

$$Y^F_{fjn} \leq \sum_{i \in I_f} Y_{ijn} \qquad \forall f, j \in J_f, n \qquad (5)$$

If we do not include constraint 5, we may obtain solutions where $Y^F_{fjn} = 1$ for a family that $\sum_{i \in I_f} Y_{ijn} = 0$; that also means $T^F_{fjn} = 0$. Note that this undesired case could be observed if the changeover cost/time for $f \rightarrow f'$ is higher than the sum of costs/times for changeovers $f \rightarrow f''$ and $f'' \rightarrow f'$.
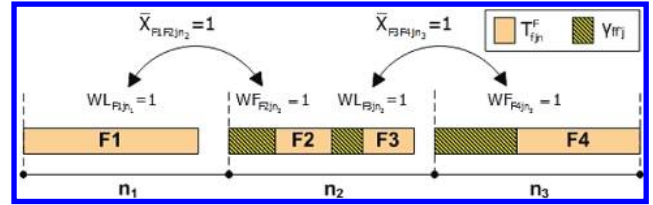

Figure 2. Family changeovers between adjacent periods.

4.3. Family Sequencing and Timing Constraints. We introduce binary variable $X_{ff'jn}$ to define the local precedence between two families f and f': $X_{ff'jn} = 1$ if family f' is processed immediately after family f in unit j. Constraints 6 and 7 ensure that, if family f is allocated on the processing unit j at period n (i.e., $Y^F_{fjn} = 1$), at most one family f' is processed before and after it, respectively. If family f is assigned first on a processing unit j (i.e., $WF_{fjn} = 1$), then it has no predecessor. Similarly, if family f is assigned last on a processing unit j (i.e., $WL_{fjn} = 1$), then it has no successor.

$$\sum_{f' \neq f, f' \in F_j} X_{f'fjn} + WF_{fjn} = Y^F_{fjn} \qquad \forall f, j \in J_f, n \qquad (6)$$

$$\sum_{f' \neq f, f' \in F_j} X_{ff'jn} + WL_{fjn} = Y^F_{fjn} \qquad \forall f, j \in J_f, n \qquad (7)$$

The correct number of immediate precedence variables is activated through constraint 8, which enforces the total number of sequenced pairs within a period to be equal to the total number of active assignments during this period minus 1:

$$\sum_{f \in F_j} \sum_{f' \neq f, f' \in F_j} X_{ff'jn} + 1 = \sum_{f \in F_j} Y^F_{fjn} \qquad \forall j, n \qquad (8)$$

To avoid sequence subcycles, we also include constraint 9, which ensures a feasible timing of the families assigned to the same processing unit:

$$C_{f'jn} \geq C_{fjn} + T^F_{f'jn} + \gamma_{ff'j} X_{ff'jn} - \omega_{jn}(1 - X_{ff'jn})$$
$$\forall f, f' \neq f, j \in (J_f \cap J_{f'}), n \qquad (9)$$

4.4. Family Changeovers Across Adjacent Periods. We introduce binary variable $\bar{X}_{ff'jn}$ to denote a changeover from family f to f' in unit j taking place at the beginning of period n. This binary variable is active only for the family f processed last in period $n-1$ (i.e., $WL_{fjn-1} = 1$) and the family f' that is processed first in period n (i.e., $WF_{f'jn} = 1$), according to constraints 10 and 11 (see Figure 2).

$$WF_{f'jn} = \sum_{f \in F_j} \bar{X}_{ff'jn} \qquad \forall f', j \in J_{f'}, n > 1 \qquad (10)$$

$$WL_{fjn-1} = \sum_{f' \in F_j} \bar{X}_{ff'jn} \qquad \forall f, j \in J_f, n > 1 \qquad (11)$$

4.5. Changeover Crossover Constraints. In most existing approaches, changeover times must begin and finish within the same period. In other words, crossovers of changeover times are not allowed. This restriction may result in suboptimal solutions. For example, in Figure 3 a better solution can be obtained if the changeover from F1 to F2 starts in period n_1 and finishes in n_2.

To model changeover crossovers, nonnegative variables $\bar{U}_{jn}$ and U_{jn} are introduced. If a changeover operation with a duration
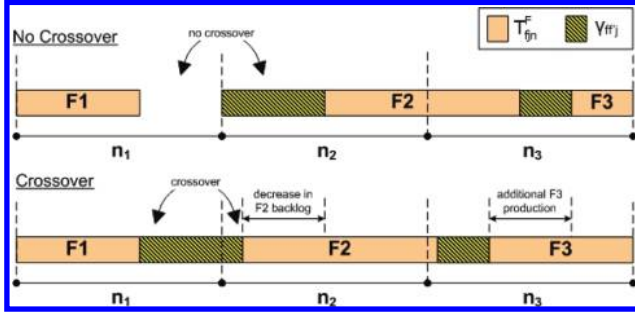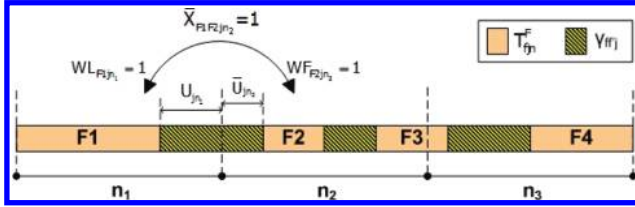
Figure 3. Crossover of changeover times.

Figure 4. Modeling of crossover of changeover times.

equal to $\gamma_{ff'j}$ starts in period $n-1$ and is continued in period n, then U_{jn-1} represents the fraction of time of the changeover operation that takes place in period $n-1$ and $\bar{U}_{jn}$ represents the fraction of time of the operation that is performed in period n (see Figure 4).

$$\bar{U}_{jn} + U_{jn-1} = \sum_{f \in F_j} \sum_{f' \in F_j, f' \neq f} \gamma_{ff'j} \bar{X}_{ff'jn} \qquad \forall j, n > 1 \quad (12)$$

4.6. Unit Production Time. The summation of the production times of families $f \in F_j$ that are processed on unit j plus the total changeover times within period n (including variables $\bar{U}_{jn}$ and U_{jn}) is constrained by the available production time ω_{jn}:

$$\bar{U}_{jn} + U_{jn} + \sum_{f \in F_j} T^F_{fjn} + \sum_{f \in F_j} \sum_{f' \in F_j, f' \neq f} \gamma_{ff'j} X_{ff'jn} \leq \omega_{jn} \qquad \forall j, n$$
$$(13)$$

where $\bar{U}_{jn=1}$ corresponds to the time point that unit j is available to begin processing any task in the first period. Notice that $\bar{X}_{ff'jn=1} = 0$ by definition.

4.7. Product Lot-Sizing Constraints. Upper and lower bounds on the production, Q_{ijn}, of product i on unit j during period n are enforced by constraint 14, where variable T_{ijn} denotes the processing time of product i on unit j in period n.

$$\rho^{min}_{ij} T_{ijn} \leq Q_{ijn} \leq \rho^{max}_{ij} T_{ijn} \qquad \forall i, j \in J_i, n \quad (14)$$

Upper and lower bounds on the processing time T_{ijn} are enforced by

$$\tau^{min}_{ijn} Y_{ijn} \leq T_{ijn} \leq \tau^{max}_{ijn} Y_{ijn} \qquad \forall i, j \in J_i, n \quad (15)$$

Note that a tight processing time upper bound τ^{max}_{ijn} can be estimated as follows:

$$\tau^{max}_{ijn} = \begin{cases} \omega_{jn} & \text{if } \sum_{n'=1}^{N} \zeta_{in}/\rho^{min}_{ij} \geq \omega_{jn} \\ \sum_{n'=1}^{N} \zeta_{in}/\rho^{min}_{ij} & \text{if } \sum_{n'=1}^{N} \zeta_{in}/\rho^{min}_{ij} < \omega_{jn} \end{cases}$$
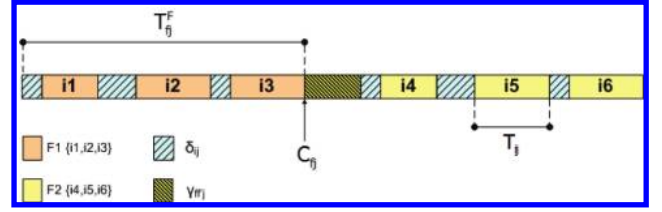

Figure 5. Family and product processing times.

Since the switchovers between products that belong to the same family ($i \in I_f$) are sequence-independent, sequencing and timing decisions regarding products can be made postoptimization without affecting the quality of solution. For example, the sequencing can be determined depending on the characteristics of the production process. The family processing time, T^F_{fjn}, is defined by (see Figure 5)

$$T^F_{fjn} = \sum_{i \in I_f} (T_{ijn} + \delta_{ij} Y_{ijn}) \qquad \forall f, j \in J_f, n \quad (16)$$

4.8. Objective Function. The optimization goal is the minimization of total inventory, backlog, changeover (inside and across periods), setup, and operating costs:

$$\min \sum_i \sum_n (\xi_{in} S_{in} + \psi_{in} B_{in}) + \sum_i \sum_{j \in J_i} \sum_n (\theta_{ij} Y_{ijn} + \lambda_{ij} Q_{ijn}) + \\ \sum_f \sum_{f' \neq f} \sum_{j \in (J_f \cap J_f)} \sum_n \phi_{ff'j} (X_{ff'jn} + \bar{X}_{ff'jn}) \quad (17)$$

4.9. Integrality and Nonegativity Constraints. The domains of decision variables are defined as follows:

$$Y_{ijn} \in \{0, 1\} \quad \text{and} \quad Q_{ijn}, T_{ijn} \geq 0 \qquad \forall i, j \in J_i, n$$

$$WF_{fjn}, WL_{fjn}, Y^F_{fjn} \in \{0, 1\} \quad \text{and} \quad T^F_{fjn}, C_{fjn} \geq 0 \qquad \forall f, j \in J_f, n$$

$$X_{ff'jn}, \bar{X}_{ff'jn} \in \{0, 1\} \qquad \forall f, f', j \in (J_f \cap J_f), n \quad (18)$$

$$B_{in}, S_{in}, P_{in} \geq 0 \qquad \forall i, n$$

$$\bar{U}_{jn} \geq 0 \qquad \forall j, n$$

The proposed MIP model, CR, consists of constraints 1−18.

4.10. Extension I: Idle Units. Note that model CR, similarly to most existing approaches, is based on the assumption that units do not remain completely idle in any period. In other words, processing units produce at least one product in each period, except maintenance periods. Generally speaking, this assumption is valid for medium to long planning periods (e.g., a production week). However, if short periods are used (e.g., a production day) to accurately model frequent intermediate due dates, idle periods may be present in an optimal solution.

To model unit idle periods, we define a dummy product $i \in I^{idle}_f$ for each family, with zero setup time and cost (i.e., $\delta_{ij} = 0$, $\theta_{ij} = 0 \,\forall\, i \in I^{idle}_f$). The processing times of dummy products are then constrained by eq 15 with $\tau^{min}_{ijn} = 0$ and $\tau^{max}_{ijn} = \omega_{jn}$. Note that if a processing unit produces only a dummy product in a production period, then this actually means that the unit remains idle during that period. Having defined a variable for idle time, we can now express constraint 13 as an equality, where $T_{ijn}, \,\forall\, i \in I^{idle}_f$ now act as slack variables. The new MIP model for idle units is CR-D.
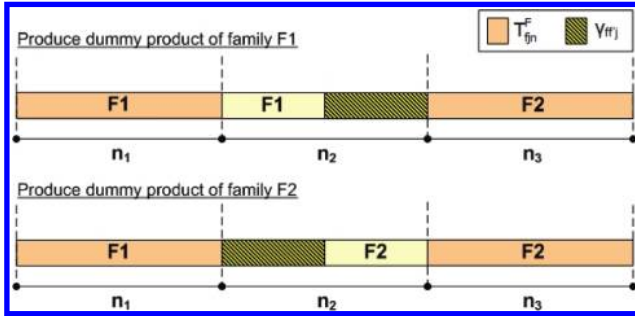
Figure 6. Modeling of changeover crossover through idle periods.
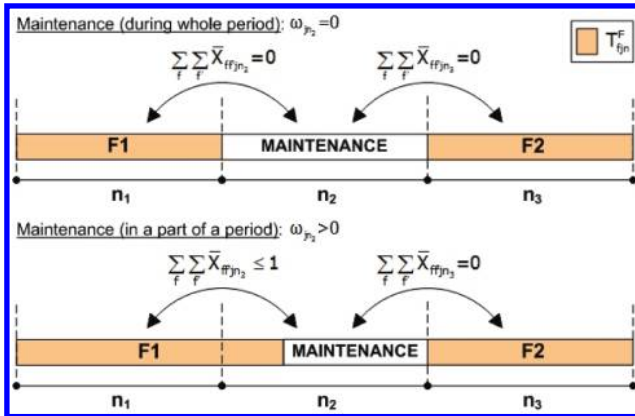

Figure 7. Modeling of maintenance activities.

Figure 6 shows an illustrative example of a single-unit production plan over three production periods, where the unit produces family F1 in period n_1, remains idle during period n_2, and produces family F2 in period n_3 (white boxes denote the imaginary production of dummy product $i \in I_f^{idle}$). Note that the two solutions are equivalent.

4.11. Extension II: Maintenance Activities. Maintenance activities in a given period can be readily addressed by fixing the corresponding changeover crossover variables at zero and modifying the available production time ω_{jn} accordingly. We assume that the maintenance activity is carried out at the end of the production period n. Our model can accommodate cases where the duration of a maintenance task is equal to the available production time (by setting $\omega_{jn} = 0$) or cases where the duration of the maintenance activity is smaller than the length of the period n. Note that if a maintenance task is performed between the production of two different families, there is no need for a changeover operation. Figure 7 illustrates our approach. Finally, note that we can also cope with maintenance tasks whose duration is greater than a planning period.

5. Applications

In this section, we discuss the application of our model to an illustrative example and a large-scale industrial case study. All formulations were solved on a Sun Ultra 4.0 Workstation with 8 GB RAM using CPLEX 11 via a GAMS 22.9[18] interface. A maximum resource limit of 300 CPU s was used for all instances. It represents the amount of time practitioners are willing to wait for a solution. This is because different scenarios must be tested before a solution is dispatched and rescheduling solutions should be generated routinely and in a timely fashion. Furthermore, we selected this short resource limit to make a fair comparison of our method to commercial tools that typically yield solutions within a few minutes.

Table 1. Illustrative Example: Changeover Times (Costs)

family	F01	F02	F03	F04	F05
F01	−	3.0 (50)	3.0 (40)	5.0 (60)	1.5 (50)
F02	5.3 (40)	−	3.0 (50)	3.0 (80)	2.0 (90)
F03	2.8 (70)	4.0 (30)	−	2.5 (80)	4.0 (30)
F04	2.4 (100)	4.0 (100)	3.0 (90)	−	3.0 (60)
F05	3.2 (30)	4.0 (50)	2.0 (50)	4.0 (70)	−

Table 2. Illustrative Example: Product Demands Per Period (kg)

product	day 1 (n_1)	day 2 (n_2)	day 3 (n_3)	day 4 (n_4)
I01	50	0	70	20
I02	0	80	10	50
I03	30	50	20	30
I04	0	10	75	10
I05	70	90	10	20
I06	65	0	75	0
I07	40	50	0	30
I08	0	45	0	0
I09	55	0	45	15
I10	10	100	30	50
I11	40	15	20	30
I12	0	95	40	30
I13	80	0	40	30
I14	0	50	0	0
I15	0	0	0	60

5.1. Illustrative Example. We consider a simple example with 15 products (I01−I15), grouped into five families (F01−F05), and three processing units (J01−J03). All products can be produced on any unit. Products are grouped into families as follows: $I_{F01} = \{I01, I02, I03\}$, $I_{F02} = \{I04, I05, I06\}$, $I_{F03} = \{I07, I08, I09\}$, $I_{F04} = \{I10, I11, I12\}$, and $I_{F05} = \{I13, I14, I15\}$. The total production horizon is 4 days and is divided into four 24-h periods. Maintenance is scheduled on units J01, J02, and J03, in periods n_2, n_3, and n_4, respectively. Processing data include the following: setup time $\delta_{ij} = 0.5$; setup cost $\theta_{ij} = 50$; operating cost $\lambda_{ij} = 0.1$; production rate $\rho_{ij}^{max} = 10$; minimum processing time $\tau_{ij}^{min} = 0.2$; inventory cost $\xi_{in} = 1$; backlog cost $\psi_{in} = 3$ for all products. Changeover times $\gamma_{ff'j}$ and costs $\phi_{ff'j}$ between product families are given in Table 1. Product demands can be found in Table 2. The processing sequence of products that belong to the same family during every period n is predetermined; specifically, it is in ascending index order.

The optimization goal is the minimization of the total cost, as defined in eq 17. The proposed MIP model obtained the optimal solution ($2,630) in 30 CPU s (see Table 3). The optimal production schedule for products and product families is shown in Figure 8. Notice the changeover crossover between families F02 and F01 on unit J01 across the boundary between the third and fourth days. Figure 9 presents the production profiles for product families and products, while Figure 10 shows the inventory and backlog profiles for every product. High inventories are observed in the first day, especially for products I05 (90 kg), I07 (50 kg), and I13 (40 kg). High backlogs are observed in the second day, for products I08 (45 kg) and I04 (10 kg), and in the third day, for products I10 (30 kg), I11 (20 kg), and I12 (15 kg).

5.2. Industrial Case Study. In this subsection, we consider a complex real-world problem in the continuous bottling stage of a beer production facility. The facility under study consists of eight processing units (J01−J08), working in parallel and producing a total of 162 products which are grouped into 22 families (F01−F22).

5.2.1. Instance I. In this instance, we study the planning− scheduling problem over a 6-week production horizon divided into six 168-h periods. The problem was solved to optimality

Table 3. Computational Results for All Problem Instances

problem instance	constraints	continuous variables	binary variables	nodes	relative gap (%)	objective function ($)	CPU s
illustrative example	1147	708	510	10 501	0.0	2,630	30
industrial case study: instance I	5192	3730	1890	97 017	0.0	235,577	193
industrial case study: instance II	8882	6269	3264	59 573	1.0	155,629	300

using model CR in less than 200 CPU s. Model and solution statistics can be found in Table 3. Figure 11 shows the Gantt chart of the optimal solution ($235,577) for product families. Note that each family block is subdivided into a number of smaller product blocks separated by setup times. The production profiles of all families in the eight units of the facility are presented in Figure 12, while Figure 13 shows the total inventory and backlog cost profiles. In the first week, we observe a high inventory cost, while backlog cost remains low during all periods. Inventories represent 34% of the total cost.

Furthermore, the solution obtained using the approach presented in this paper was compared against the solution that was found, dispatched, and executed in practice using a combination of in-house and commercial tools. The main cost components of the two solutions are compared in Figure 14. Clearly, our solution is substantially better. In particular, inventory cost is 78% lower than the inventory cost of the executed solution, and backlog cost is less than 5% of the executed solution. Also, changeover and setup costs have been reduced by more than 80% and 25%, respectively (see the table in the bottom of Figure 14). The results of this case study indicate that the proposed framework can in fact be used to address real-world problems. It is computationally efficient and yields solutions of very good quality. Given the significant improvement over the practiced methods, our formulation is currently incorporated into and tested using the tools currently employed to generate detailed production plans.
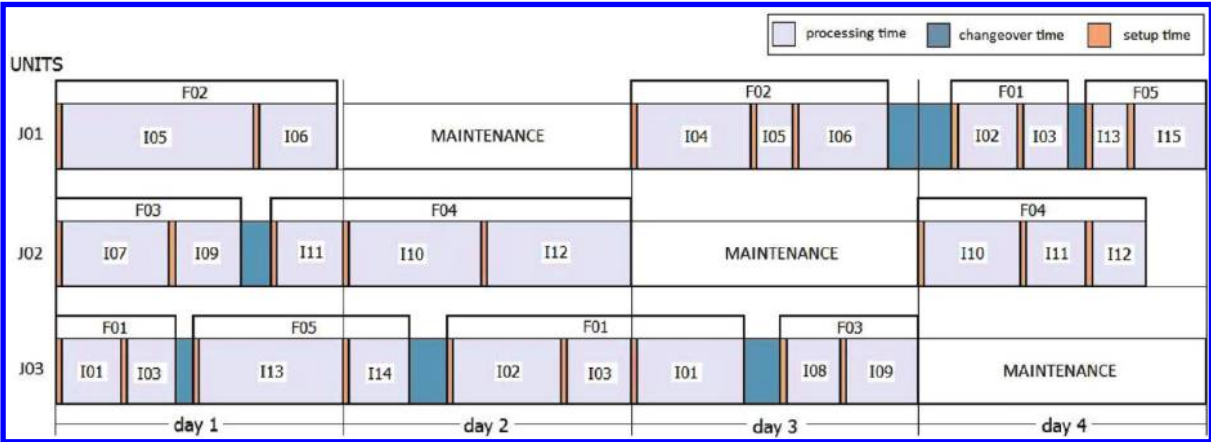

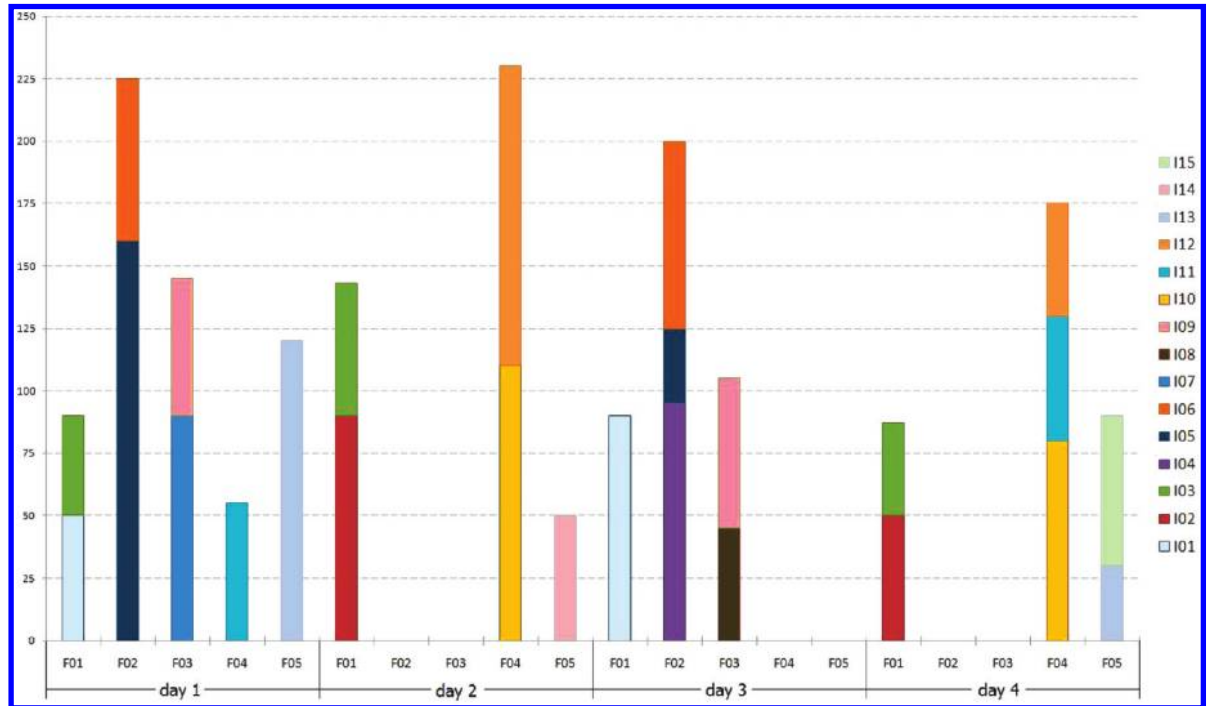**Figure 8.** Illustrative example: Gantt chart of optimal solution.

Figure 9. Illustrative example: production profiles of families and products (kg).
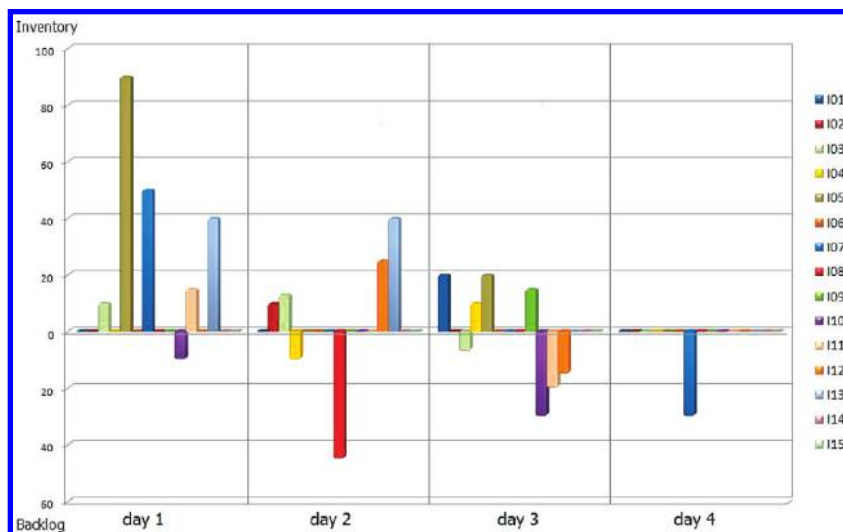
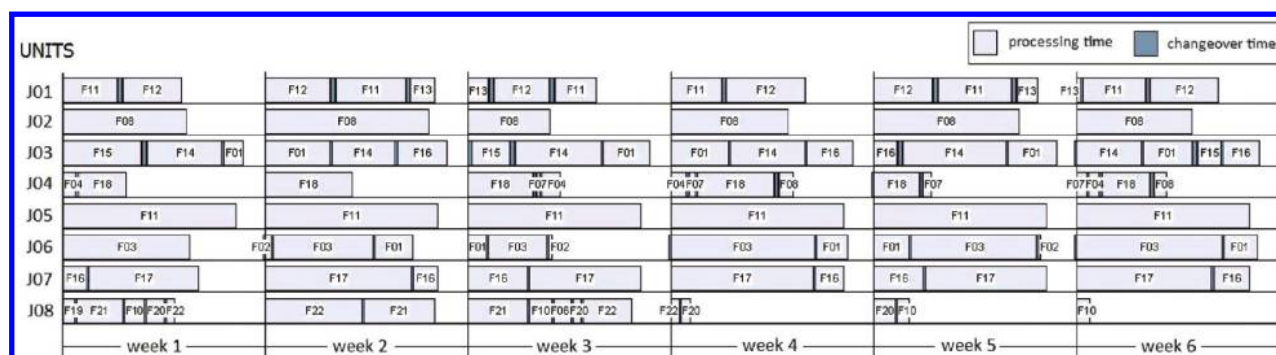**Figure 10.** Illustrative example: product inventory and backlog profiles (kg).

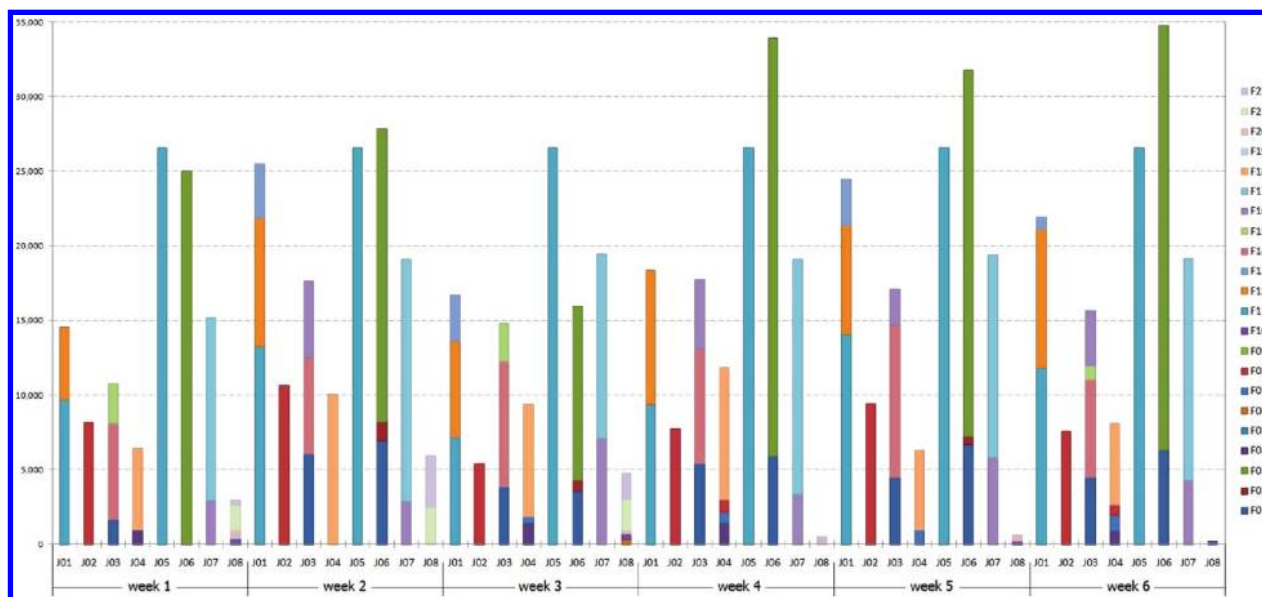**Figure 11.** Industrial case study, instance I: family Gantt chart of optimal solution.

Figure 12. Industrial case study, instance I: production profile for families (kg).

5.2.2. Instance II. In this instance, we discuss a problem in the same processing facility but over a planning horizon of 4 weeks partitioned into seven 24-h (week 1) and three 168-h planning periods (weeks 2−4). In other words, we consider daily orders during the first week, and weekly orders during the next three weeks. This partitioning represents industrial practice, where orders in the near future are known with certainty and are treated separately, while orders in later periods are aggregated. The consideration of such small planning periods leads to larger formulations that are harder to solve to optimality, but are necessary to avoid unmet demand. Furthermore, good solutions to these problems often require changeover crossovers,
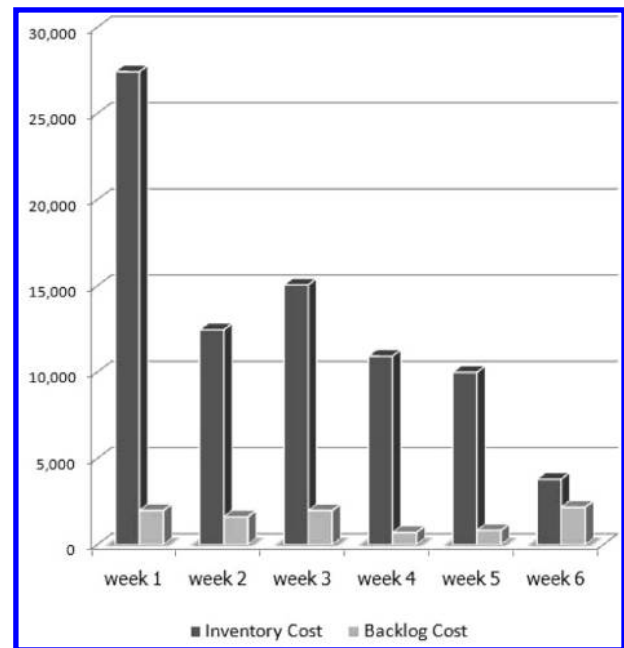
Figure 13. Industrial case study, instance I: total inventory and backlog cost profiles ($).

Figure 16. Industrial case study, instance II: cost analysis ($).

as well as idling of units for more than one (short) planning period. To address this instance, we developed a formulation that combines model CR-D for the first seven days, allowing units to remain idle over 1 day, and model CR for the three 7-day production periods.
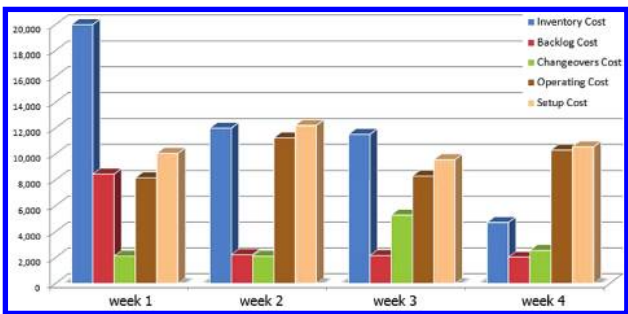
The model and solution statistics of the resulting MIP model can be found in Table 3. The best solution obtained within 300 CPU s has a total cost of $155,629 and has an optimality gap equal to 1%. The solution obtained using the proposed framework is again significantly better than the solution found by commercial tools. The production profile of all families is shown in Figure 15, while a cost analysis for every week can be found in Figure 16, where for week 1 we present the aggregated costs for day periods 1−7. It is worth noting that the consideration of 24-h planning periods during the first week results in higher inventory and backlog costs because daily demands are harder to meet. Inventories represent 31% of the total cost (vs 34% in instance I), but backlog costs have increased to 10% from 4% in instance I. Finally, note that we do observe idle 24-h periods during the first week.
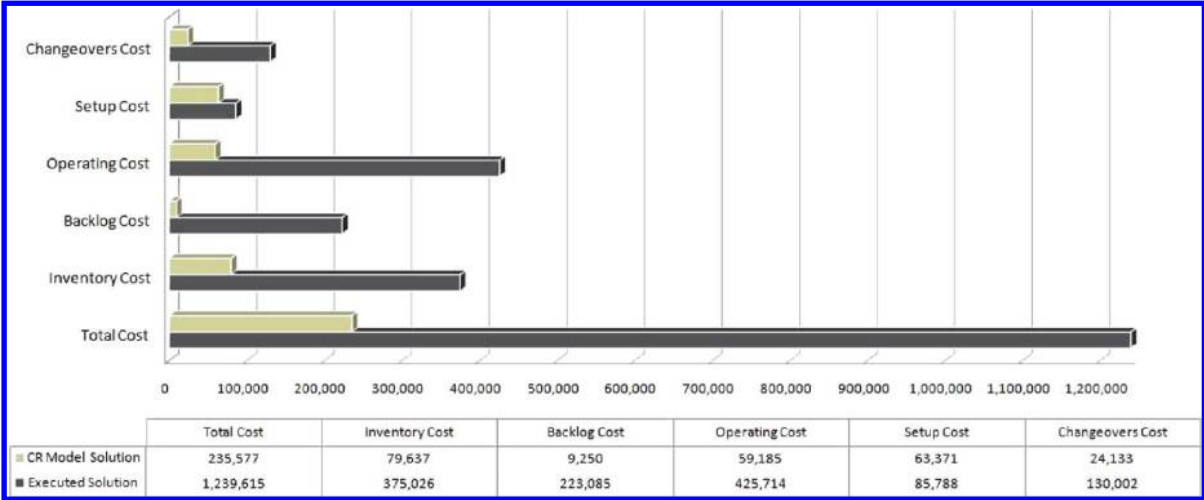

	Total Cost	Inventory Cost	Backlog Cost	Operating Cost	Setup Cost	Changeovers Cost
CR Model Solution	235,577	79,637	9,250	59,185	63,371	24,133
Executed Solution	1,239,615	375,026	223,085	425,714	85,788	130,002

Figure 14. Industrial case study, instance I: comparison of solutions obtained by the proposed MIP model and the tools currently used in practice ($).
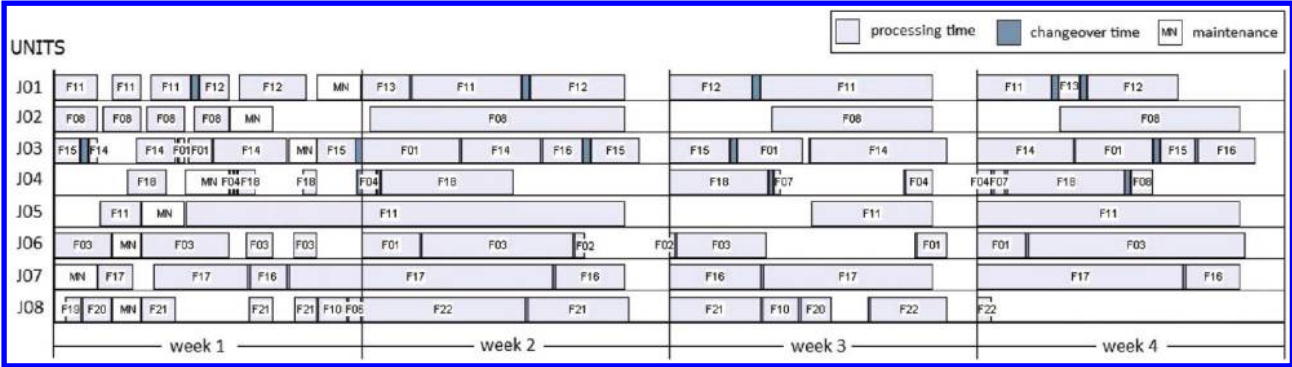

Figure 15. Industrial case study, instance II: family Gantt chart of best solution.

6. Conclusions

In this paper, we presented a mixed integer programming (MIP) formulation for the production planning and scheduling of single-stage continuous processes with product families, a type of production system that appears in a number of different production environments and industrial sectors. Our approach combines a discrete-time partitioning of the planning horizon to account for the major planning decisions (production targets, shipments, and inventory levels) with a continuous-time treatment of detailed scheduling decisions within each planning period. Furthermore, at the scheduling level, it combines a *precedence-based* approach to correctly enforce sequencing constraints among product families with a *lot-sizing-like* approach to account for production time constraints for individual products. Our approach addresses appropriately aspects such as changeover carryover and crossover, thereby leading to solutions with higher utilization of resources. Also, it is not based upon the assumption that processing units cannot remain idle during a production period, thereby allowing us to partition the planning horizon into smaller periods, which in turn results in better solutions. Importantly, the integration of these approaches leads to computationally effective MIP models. Very good solutions to problems with hundreds of products can be obtained within 5 CPU min, while optimal solutions can also be found in a reasonable time. Furthermore, the proposed formulation yields solutions which are substantially better than the ones obtained using commercial tools, suggesting that MIP methods can be used to address problems of practical interest.

Acknowledgment

Nomenclature

Indices/Sets

$i \in I$ = products
$f, f' \in F$ = families (product families)
$j \in J$ = processing units
$n, n' \in N$ = production periods

Subsets

F_j = families that can be processed in unit j
I_f = products that belong to family f
I_f^{idle} = dummy product for family f (one per family)
I_j = products that can be processed in unit j
J_f = processing units that can process family f
J_i = processing units that can process product i

Parameters

$\gamma_{ff'j}$ = changeover time between families f and f' in unit j
δ_{ij} = setup time of product i in unit j
ζ_{in} = demand of product i at time n
θ_{ij} = setup cost of product i in unit j
λ_{ij} = operating cost of product i in unit j

ξ_{in} = holding cost of product i in period n
ρ_{ij}^{max} = maximum production rate of product i in unit j
ρ_{ij}^{min} = minimum production rate of product i in unit j
τ_{ij}^{max} = maximum processing time of product i in unit j
τ_{ij}^{min} = minimum processing time of product i in unit j
$\phi_{ff'j}$ = changeover cost between families f and f' in unit j
ψ_{in} = backlog cost of product i in period n
ω_{jn} = available production time in unit j in period n

Continuous Variables

B_{in} = backlog of product i at time n
C_{fjn} = completion time for family f in unit j in period n
P_{in} = total produced amount of product i in period n
Q_{ijn} = produced amount of product i in unit j during period n
S_{in} = inventory of product i at time n
T_{ijn} = processing time for product i in unit j in period n
T_{fjn}^F = processing time for family f in unit j in period n
U_{jn} = time within period n consumed by a changeover operation that will be completed in the next period on unit j
$\bar{U}_{jn}$ = time within period n consumed by a changeover operation that started in the previous period on unit j

Binary Variables

WF_{fjn} = 1 if family f is assigned first to unit j in period n
WL_{fjn} = 1 if family f is assigned last to unit j in period n
$X_{ff'jn}$ = 1 if family f is processed exactly before f' in period n in unit j
$\bar{X}_{ff'jn}$ = 1 if family f in period $n-1$ is immediately followed from family f' in period n on unit j
Y_{ijn} = 1 if product i is assigned to unit j in period n
Y_{fjn}^F = 1 if family f is assigned to unit j in period n

Literature Cited

(1) Méndez, C. A.; Cerdá, J.; Grossmann, I. E.; Harjunkoski, I.; Fahl, M. Review: State-of-the-art of optimization methods for short-term scheduling of batch processes. *Comput. Chem. Eng.* **2006**, *30*, 913–946.

(2) Shah, N. Single and multisite planning and scheduling: current status and future challenge. *Proceedings of the Third International Conference of the Foundations of Computer-Aided Process Operations*; Snowbird, Utah, 1998; pp 75−90.

(3) Kallrath, J. Planning and scheduling in the process industry. *OR Spectrum* **2002**, *24*, 219–250.

(4) Maravelias, C. T.; Sung, C. Integration of production planning and scheduling: Overview, challenges and opportunities. *Comput. Chem. Eng.* **2009**, *33*, 1919–1930.

(5) Pochet, Y.; Wolsey, L. A. *Production Planning by Mixed Integer Programming*; Springer Series in Operations Research and Financial Engineering; Springer-Verlag: Secaucus, NJ, 2006.

(6) Chen, P.; Papageorgiou, L. G.; Pinto, J. M. Medium-term planning of single-stage single-unit multiproduct plants using a hybrid discrete/continuous-time MILP model. *Ind. Eng. Chem. Res.* **2008**, *47*, 1925–1934.

(7) Liu, S.; Pinto, J. M.; Papageorgiou, L. G. A TSP-based MILP model for medium-term planning of single-stage continuous multiproduct plants. *Ind. Eng. Chem. Res.* **2008**, *47*, 7733–7743.

(8) Liu, S.; Pinto, J. M.; Papageorgiou, L. G. MILP-based approaches for medium-term planning of single-stage continuous multiproduct plants with parallel units. *Comput. Manage. Sci.* **2009**, *7*, 407–435.

(9) Erdirik-Dogan, M.; Grossmann, I. E. A decomposition method for the simultaneous planning and scheduling of single-stage continuous muliproduct plants. *Ind. Eng.Chem. Res.* **2006**, *45*, 299–315.

(10) Erdirik-Dogan, M.; Grossmann, I. E. Simultaneous planning and scheduling of single-stage multi-product continuous plants with parallel lines. *Comput. Chem. Eng.* **2008**, *32*, 2664–2683.

(11) Inman, R. R.; Jones, P. C. Decomposition for scheduling flexible manufacturing systems. *Oper. Res.* **1993**, *41*, 608–617.

(12) Grunow, M.; Günther, H.-O.; Lehmann, M. Campaign planning for multi-stage batch processes in the chemical industry. *OR Spectrum* **2002**, *24*, 281–314.

(13) Günther, H.-O.; Grunow, M.; Neuhaus, U. Realizing block planning concepts in make-and-pack production using MILP modelling and SAP APO. *Int. J. Prod. Res.* **2006**, *44*, 3711–3726.

(14) Kopanos, G. M.; Puigjaner, L.; Georgiadis, M. C. Optimal production scheduling and lot-sizing in dairy plants: The yogurt production line. *Ind. Eng. Chem. Res.* **2010**, *49*, 701–718.

(15) Karmarkar, U. S.; Schrage, L. The Deterministic Dynamic Product Cycling Problem. *Oper. Res.* **1985**, *33*, 326–345.

(16) Sahinidis, N. V.; Grossmann, I. E. MINLP model for cyclic multiproduct scheduling on continuous parallel lines. *Comput. Chem. Eng.* **1991**, *15*, 85–103.

(17) Sung, C.; Maravelias, C. T. A mixed-integer programming formulation for the general capacitated lot-sizing problem. *Comput. Chem. Eng.* **2008**, *32*, 244–259.

(18) Brooke, A.; Kendrik, D.; Meeraus, A.; Raman, R.; Rosenthal, R. E. *GAMS—A User's Guide*; GAMS Development Corp.: Washington, DC, 1998.