A heuristic decomposition method for the short-term planning of continuous plants

Christoph Schwindt, Sascha Herrmann
Institute for Management and Economics
Clausthal University of Technology
D-38678 Clausthal-Zellerfeld, Germany
Tel: +49 5323 72 7615, Fax: +49 5323 72 7698
{christoph.schwindt,sascha.herrmann}@tu-clausthal.de

Norbert Trautmann
Department of Business Administration
University of Bern
CH-3012 Bern, Switzerland
Tel: +41 31 631 5325, Fax: +41 31 631 8421
trautmann@ifm.unibe.ch

Abstract

In this paper we consider the short-term production planning problem of multipurpose continuous plants. This problem quite naturally decomposes into a planning problem of optimizing the operating conditions and processing times of the continuous tasks and a scheduling problem, which consists in allocating processing units, input materials, and storage space over time to the resulting operations. The planning problem can be formulated as a continuous nonlinear programming problem of moderate size. Due to constraints on material availability and storage capacity for intermediate products, for solving the scheduling problem classical schedule-generation schemes cannot be applied. That is why we use a new two-phase approach dealing with the two types of constraints separately.

1. Introduction

A multipurpose continuous production plant consists of several processing units and storage facilities for intermediate products, which are linked by flexible hoses or pipelines. Final products are produced through a sequence of continuous tasks being executed on individual processing units. A processing unit must be cleaned between the executions of different tasks, the cleaning times generally being sequence-dependent. Each task may be executed on alternative processing units, and for each task the processing time, the production rate, as well as the proportions of the input and output products may be chosen within prescribed intervals. Whereas certain intermediate products can be stocked in dedicated storage facilities of finite capacity, others are chemically instable and must be consumed instantly. The execution of a task on a processing unit during a specified processing time and with specified production rate and input and output proportions is referred to as an operation. Given a set of primary requirements for the final products of some family, the short-term production planning problem consists in first, generating an appropriate set of operations (planning problem) and second, scheduling the operations on the processing units of the plant (scheduling problem). Since the plant can only be reconfigured for processing the next product family when all operations have been completed, we assume that the objective is to minimize the makespan needed for satisfying the given primary requirements.

A significant body of chemical engineering research has been focused on the short-term planning of batch plants. Much less work has been reported on continuous process scheduling even though continuous processes constitute an important component of process industries. Classical approaches to continuous process scheduling assumed the demand for final products to be constant over time (see e.g., Sahinides and Grossmann 1991). More recently, different types of MINLP and MILP continuous-time formulations have been developed for the general short-term production planning of continuous plants with demands at discrete points in time (such models have for example been devised by Ierapetritou and Floudas 1998, Mockus and

Reklaitis 1999, and Méndez and Cerdá 2002). A relaxation-based branch-and-bound algorithm for solving the scheduling problem for given set of operations has been proposed by Neumann et al. (2005). Constraint propagation techniques for the scheduling of continuous material flows can be found in Sourd and Rogerie (2002).

In contrast to the monolithic MINLP and MILP models, we follow a hierarchical approach, with the planning problem at the top level and the scheduling problem at the base level. This heuristic decomposition of the problem allows us to cope with instances of practical size within a reasonable amount of computing time.

2. The planning problem

In this section we consider the planning problem in more detail. Let $\mathcal{T}$ be the set of all tasks τ and let $\mathcal{P}$ be the set of all raw materials, intermediate products, and final products π under consideration. For each task $\tau \in \mathcal{T}$ we have to determine the processing time p_τ, the production rate γ_τ, and the input and output proportions $\alpha_{\tau\pi}$ of all products $\pi \in \mathcal{P}$ consumed or produced, respectively, during the execution of task τ, where we establish the convention that $\alpha_{\tau\pi} < 0$ for input products π. By $\mathcal{P}_\tau^-$ and $\mathcal{P}_\tau^+$ we denote the sets of all input or output products of task τ. Symmetrically, $\mathcal{T}_\pi^- := \{\tau \in \mathcal{T} \mid \pi \in \mathcal{P}_\tau^-\}$ and $\mathcal{T}_\pi^+ := \{\tau \in \mathcal{T} \mid \pi \in \mathcal{P}_\tau^+\}$ are the sets of all tasks consuming or producing product π. Moreover, let $\mathcal{P}^i$ be the set of all intermediate products and $\mathcal{P}^p \subseteq \mathcal{P}^i$ be the set of all perishable intermediate products. For each product $\pi \in \mathcal{P}$ a primary requirement ρ_π, including an unavoidable residual stock minus the initial stock, as well as the capacity σ_π of the storage facility for π are given, where $\sigma_\pi = 0$ if $\pi \in \mathcal{P}^p$. For simplicity, we assume that all alternative processing units on which a given task $\tau \in \mathcal{T}$ can be executed are identical. As a consequence, the assignment of units to tasks may be deferred to the scheduling phase, and the planning problem (PP) can be formulated as follows:

$$
\text{(PP)}\begin{cases}
\text{Minimize} & \sum_{\tau\in\mathcal{T}} p_\tau & & \\
\text{subject to} & \underline{\alpha}_{\tau\pi} \le \alpha_{\tau\pi} \le \overline{\alpha}_{\tau\pi} & (\tau\in\mathcal{T}, \pi\in\mathcal{P}_\tau^- \cup \mathcal{P}_\tau^+) & (1)\\
& \underline{p}_\tau \le p_\tau \le \overline{p}_\tau & (\tau\in\mathcal{T}) & (2)\\
& \underline{\gamma}_\tau \le \gamma_\tau \le \overline{\gamma}_\tau & (\tau\in\mathcal{T}) & (3)\\
& \sum_{\pi\in\mathcal{P}_\tau^+} \alpha_{\tau\pi} = -\sum_{\pi\in\mathcal{P}_\tau^-} \alpha_{\tau\pi} = 1 & (\tau\in\mathcal{T}) & (4)\\
& \rho_\pi \le \sum_{\tau\in\mathcal{T}_\pi^- \cup \mathcal{T}_\pi^+} \alpha_{\tau\pi}\gamma_\tau p_\tau \le \rho_\pi + \sigma_\pi & (\pi\in\mathcal{P}) & (5)\\
& \alpha_{\tau\pi}\gamma_\tau = -\alpha_{\tau'\pi}\gamma_{\tau'} & (\pi\in\mathcal{P}^p, \tau\in\mathcal{T}_\pi^+, \tau'\in\mathcal{T}_\pi^-) & (6)\\
& \sum_{\tau\in\mathcal{T}_\pi^+} p_\tau = \sum_{\tau\in\mathcal{T}_\pi^-} p_\tau & (\pi\in\mathcal{P}^p) & (7)\\
& \left(\frac{1}{|\mathcal{T}_\pi^+|}\sum_{\tau\in\mathcal{T}_\pi^+} \alpha_{\tau\pi}\gamma_\tau\right)\left(\sum_{\tau\in\mathcal{T}_\pi^+} p_\tau - \sum_{\tau\in\mathcal{T}_\pi^-} p_\tau\right) \le \sigma_\pi & (\pi\in\mathcal{P}^i\setminus\mathcal{P}^p) & (8)\\
& \left(\frac{1}{|\mathcal{T}_\pi^+|}\sum_{\tau\in\mathcal{T}_\pi^+} \alpha_{\tau\pi}\gamma_\tau + \frac{1}{|\mathcal{T}_\pi^-|}\sum_{\tau\in\mathcal{T}_\pi^-} \alpha_{\tau\pi}\gamma_\tau\right)\sum_{\tau\in\mathcal{T}_\pi^+} p_\tau \le \sigma_\pi & (\pi\in\mathcal{P}^i\setminus\mathcal{P}^p) & (9)
\end{cases}
$$

The goal of the planning phase consists in defining the operating conditions and processing times in such a way that the workload to be scheduled during the scheduling phase is minimized. Constraints (1)–(3) define the feasible domains of input and output proportions, processing times, and production rates. The mass balance constraint (4) says that for each task the amount of products consumed coincides with the amount of products produced. Constraint (5) ensures that the final inventories of the products after the execution of all tasks are sufficiently large to meet the primary requirements and that the residual stocks after having satisfied all demands do not exceed the storage capacities. Constraint (6) guarantees that each perishable product $\pi \in \mathcal{P}^p$ produced by some task τ can be simultaneously consumed at the same rate by any consuming task $\tau' \in \mathcal{T}_\pi^-$. Constraint (7) requires that the total production time of a perishable product π equals the total consumption time, which together with (6) implies that the amount of π produced coincides with the amount consumed. Inequalities (8) and (9) ensure that the operating conditions and processing times are chosen in such a way that there exists a production schedule satisfying the storage-capacity constraints for the storable intermediate products, where we suppose that all tasks producing or consuming, respectively, the same intermediate product are executed one after another. For the case where some or all of those tasks can be processed in parallel, appropriate relaxations of (8) and (9) can be used.

By introducing the new variables $\xi_{\tau\pi} := \alpha_{\tau\pi}\gamma_\tau$ ($\tau \in \mathcal{T}$, $\pi \in \mathcal{P}_\tau^- \cup \mathcal{P}_\tau^+$), problem (PP) can be transformed into a nonlinear programming problem with linear objective function, linear constraints (1)–(4), (6), (7), and bilinear constraints (5), (8), and (9).

3. The scheduling problem

As a result of the planning phase we have obtained a set O of operations i of durations p_i, which must be scheduled on the processing units subject to material-availability and storage-capacity constraints. The scheduling problem can be modeled as a resource-constrained project scheduling problem with renewable resources $k \in \mathcal{R}^\rho$, continuous cumulative resources $l \in \mathcal{R}^\gamma$, and sequence-dependent changeover times between operations. By S_i we denote the start time of operation i, and $S = (S_i)_{i\in O}$ is the production schedule sought.

We group identical processing units to a renewable resource $k \in \mathcal{R}^\rho$ whose capacity R_k is equal to the number of units in the group. If operation i is executed on a unit of resource k, the resource requirement is $r_{ik} = 1$, otherwise we have $r_{ik} = 0$. Between consecutive operations $i,j \in O$ that are carried out on the same unit of resource k a sequence-dependent changeover time arises for cleaning the unit. By $r_k(S)$ we denote the minimum number of units of resource k needed to implement schedule S with respect to all assignments of operations to units. Number $r_k(S)$ can be computed efficiently using network flow algorithms (see e.g., Schwindt 2005, Sect. 5.2).

In addition, we associate a continuous cumulative resource $l \in \mathcal{R}^\gamma$ with each product $\pi \in \mathcal{P}$, $\underline{R}_l := 0$ and $\overline{R}_l := \sigma_\pi$ being the minimum and maximum inventory levels of l. The (total) requirement of operation i belonging to task τ for resource l stocking product π equals $r_{il} := \alpha_{\tau\pi}\gamma_\tau p_i$. Let $x_i(S,t)$ be the portion of operation $i \in O$ that has been processed by time t given schedule S. The inventory of continuous cumulative resource l at time t is $r_l(S,t) := \Sigma_{i\in O} r_{il} x_i(S,t)$. The scheduling problem (SP) now reads as follows:

$$
\text{(SP)} \begin{cases}
\text{Minimize} & \max_{i\in O}(S_i + p_i) & & \\
\text{subject to} & r_k(S) \le R_k & (k \in \mathcal{R}^\rho) & (10)\\
& \underline{R}_l \le r_l(S,t) \le \overline{R}_l & (l \in \mathcal{R}^\gamma) & (11)\\
& S_i \ge 0 & (i \in O) & (12)
\end{cases}
$$

The objective function corresponds to the production makespan to be minimized. Constraint (10) guarantees that there exists an assignment of the operations to the processing units such that for each renewable resource k the number of units used jointly does not exceed the resource capacity at any point in time. The inequalities in constraint (11) correspond to the material-availability and storage-capacity constraints, and (12) are the nonnegativity conditions for start times S_i.

For solving scheduling problem (SP) we use a two-phase priority-rule based heuristic. A preliminary version of this method for the case of batch production can be found in Schwindt and Trautmann (2004). At first, we perform a preprocessing step where we apply a number of constraint propagation techniques providing temporal constraints of type $S_j - S_i \ge \delta_{ij}$ between the start times of certain operations $i,j \in O$. It is customary to represent those temporal constraints by an operation-on-node network N. In the first phase of the priority-rule based method, we relax the storage-capacity constraints. Using a serial schedule-generation scheme, in each iteration we schedule an eligible operation j at the earliest point in time at which renewable resource k with $r_{jk} = 1$ and all input products are available during the entire execution time. An operation j is eligible if two conditions are met. First, all predecessors i of j with respect to a specified precedence order $\prec$ must have been scheduled. A classical precedence order is the distance order where $i \prec j$ if the temporal constraints imply $S_i \leq S_j$ but not $S_j \leq S_i$ (see Neumann et al. 2003, Sect. 1.4). It proves to be expedient to modify the precedence order such that all operations from a strong component in N are scheduled one after another. The second condition says that the remaining storage capacities after the completion of all operations i scheduled in previous iterations must suffice to stock the output products of operation j.

After the termination of the first phase we then introduce, for each intermediate product, appropriate temporal constraints between operations producing and consuming, respectively, the product according to a FIFO policy. Those temporal constraints are satisfied by schedule S and ensure that no shortage of input products can occur during the execution of any operation. In the second phase, we re-perform the scheduling of the operations, starting each operation at the earliest point in time where the temporal constraints are satisfied and where during the execution of the operation the processing unit and storage capacity for all output products are available.

Sometimes it may happen that due to temporal constraints between operations i already scheduled and the operation j selected, the latest time-feasible start time of j is strictly smaller than the earliest resource-feasible start time of j. Then no feasible start time can be found for operation j, and the current partial schedule cannot be extended to a feasible schedule. Since already the problem of finding a feasible schedule is NP-hard, this kind of deadlock cannot be avoided in a schedule-construction procedure. To resolve the deadlock, we perform the following unscheduling step. We determine the set $\mathcal{U}$ of all scheduled operations i that must be delayed when increasing the latest start time of j, we increase the earliest start times of operations $i \in \mathcal{U}$ beyond times S_i, and we restart the scheduling procedure with the modified earliest start times.

References

[1] Ierapetritou, M.G., Floudas, C.A. (1998) Effective continuous-time formulation for short-term scheduling: 2. Continuous and semicontinuous processes. *Industrial Engineering Chemistry Research*, 37, 4360–4374.

[2] Méndez, C.A., Cerdá, J. (2002). An efficient MILP continuous-time formulation for short-term scheduling of multiproduct continuous facilities. Computers and Chemical Engineering, 26, 687–695.

[3] Mockus, L., Reklaitis, G.V. (1999). Continuous time representation approach to batch and continuous process scheduling: 1. MINLP formulation. Computers and Chemical Engineering, 26, 687–695.

[4] Neumann, K., Schwindt, C., Zimmermann, J. (2003). Project Scheduling with Time Windows and Scarce Resources. Springer, Berlin.

[5] Neumann, K., Schwindt, C., Trautmann, N. (2005). Scheduling of continuous and discontinuous material flows with intermediate storage restrictions. European Journal of Operational Research, 165, 495–509.

[6] Sahinides, N.V., Grossmann, I.E. (1991). MINLP model for cyclic multiproduct scheduling on continuous parallel lines. Computers and Chemical Engineering, 15, 85–103

[7] Schwindt, C. (2005). Resource Allocation in Project Management. Springer, Berlin.

[8] Schwindt, C., Trautmann, N. (2004). A priority-rule based method for batch production scheduling in the process industries. In: Ahr, D., Fahrion, R., Oswald, M., Reinelt, G. (eds.) Operations Research Proceedings 2003. Springer, Berlin, pp. 111–118.

[9] Sourd, F., Rogerie, J. (2002). Continuous filling and emptying of storage systems in constraint-based scheduling. Proceedings of the Eighth International Workshop on Project Management and Scheduling, Valencia, pp. 342–345.