Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Short-term Multiproduct Batch Scheduling Considering Storage Features ⋆

Ouyang Wu [*,**] **Giancarlo Dalle Ave** [***,****]
Iiro Harjunkoski [***] **Ala Bouaswaig** [*] **Stefan Marco Schneider** [*]
Matthias Roth [*] **Lars Imsland** [**]

[*] *Automation Technology, BASF SE, 67056 Ludwigshafen, Germany*
[**] *Department of Engineering Cybernetics, NTNU, 7491 Trondheim, Norway*
[***] *ABB Corporate Research Germany, 68526 Ladenburg, Germany*
[****] *Department of Biochemical & Chemical Engineering, TU Dortmund, 44221 Dortmund, Germany*

Abstract: In this paper, modeling of storage constraints and material transfer are considered for short-term batch scheduling. The key features of storage and quality checks are inspired from a case study of a multiproduct batch plant. The case study presents two strategies for assigning batch orders to each individual storage tank during batch production, which are modeled in the proposed scheduling formulations as two scenarios of storage policies. A continuous-time MILP formulation is applied for the assignment and sequencing of batches in the multistage processes. The proposed approach is tested using the case study problems, and the computational results illustrate the performance of the scheduling formulations.

Keywords: Storage policies, Multiproduct batch production, Precedence-based MILP models

1. INTRODUCTION

Batch scheduling is an important topic in the process industries, and it is relevant for a variety of industrial batch plants. Characteristics of batch processes are often modeled in mixed-integer linear programming (MILP) formulations of scheduling problems. Méndez et al. (2006) summarizes main features of batch scheduling models such as multiple stages with parallel units, inventory storage policies, batching of orders, demand patterns, etc. Equipment degradation is another key feature often considered in integrated scheduling of production and maintenance (Biondi et al., 2017; Aguirre and Papageorgiou, 2018; Wu et al., 2019; Dalle Ave et al., 2019).

Storage features in batch scheduling are modeled via a variety of formulations. Sundaramoorthy and Maravelias (2008) considered constraints of storage in simultaneous batching and scheduling of multiproduct multistage processes; the storage policies include capacity of storage such as number, size of storage vessels, and timing constraints on the storage time and waiting time. The MILP problem is formulated using a precedence-based method. Precedence-based models are widely applied for multistage batch processes, in which global or immediate precedence variables define the precedence relations of any two batch runs resulting in an MILP set of sequencing constraints (Méndez and Cerdá, 2003; Gupta and Karimi, 2003). Kilic et al. (2011) proposed storage constraints in a discrete-time batch scheduling MILP formulation; by using the state-task-network formulation proposed by Kondili et al. (1993), material flows are defined to model the storage vessels explicitly.

This paper considers storage and material transfer features for the scheduling of batch processes, which is inspired from a real multiproduct batch case study. In the case study, batches of products are produced sequentially in the stages of production, and tanks store batches of the same product for product quality checks. To model the assignment and sequencing of batch products in the storage tanks, a concept of *groups* is introduced to represent the union of batches in individual tanks for the check. Constraints for the assignment of groups in the storage tanks are proposed, and two scenarios of storage policies for the multigrade products are considered.

The main contributions of this paper are the proposal of new scheduling formulations that consider features of storage and material transfer outlined in the case study example. In this paper, the work in Wu et al. (2019, 2020) is extended to consider the assignment of batches to storage tanks; Wu et al. (2019, 2020) proposed precedence-based scheduling formulations for the short-term scheduling of multiproduct batch processes, in which sequence-dependent degradation is modeled for condition-aware production scheduling; Wu et al. (2020) further considered scheduling of integrated production and maintenance. This paper models additional constraints on the storage and material transfer operations in the case study and integrates them in the precedence-based scheduling formulation from Wu et al. (2019, 2020). Since the focus is the modeling of storage and sequencing of groups in tanks, other aspects of the case study mentioned in Wu

⋆ Financial support is gratefully acknowledged from the Marie Skłodowska Curie Horizon 2020 EID-ITN project "PRONTO", Grant agreement No 675215.

et al. (2019, 2020), such as the degradation evolution and the scheduling of maintenance, are not considered and included in the formulation, but could in principle be added. Furthermore, the scheduling formulations are tested on problem instances from the case study. The computational results and resulting optimal solutions are compared for two storage policy scenarios, which also provides scheduling information for the the actual operations in the plant.

The rest of this paper is structured as follows. Section 2 describes the case study and the features that are considered in the scheduling formulations. Next, nomenclature is presented in Section 3 providing descriptions of symbols in the scheduling models that are demonstrated in Section 4. Section 5 presents and discusses the results of the computational tests of the proposed scheduling formulation. Conclusions are drawn in Section 6.

2. PROBLEM DESCRIPTION

This section presents the features of batch scheduling that are considered in the case study example.

2.1 Multistage multiproduct batch process

The batch polymerization process considered consists of three stages, with the possibility of parallel units at each stage as Fig. 1 shows. All batches must be processed in one unit of each stage and visit the stages following the same sequence: raw materials are added to vessel U1 for monomer make-up according to recipes in stage L1; the prepared monomer is then fed to a homogenizer where it is mixed with oil to form monomer emulsion and then directly transferred into a batch reactor in stage L2; batch products become ready when the polymerization is finished in either reactors U2 or U3, and the batch is transferred to one of the storage tanks in stage L3 for final quality check.

In the case study, the number and sizes of batches are usually known a priori before scheduling of batch runs. Demands of multigrade products are batched into many orders that follow different types of recipes as Fig. 1 shows. Each recipe stands for a unique product grade; orders of the same recipe have the same batch size and are identical with respect to operations during each batch run.

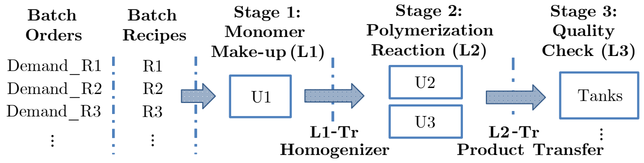

Fig. 1. Process topology of the case study

2.2 Material transfer between stages

In this problem, material transfer time and structure must be considered when scheduling. Figure 1 shows two types of material transfer: L1-Tr denotes monomer transfer from vessel U1 to one of the reactors through the homogenizer; L2-Tr denote product transfer from reactors U2 or U3 to one of the storage tanks.

Material transfer between two neighboring stages are performed through the connecting pipes. However, the number of pipes can be less than the number of units in the stage leading to shared pipes of multiple units. As a result, material transfer for units in the same stage needs to be scheduled to avoid overlapping in the pipes. In the example, it is considered that two reactors use only a single pipe to transfer products to storage.

2.3 Product quality check at storage tanks

Products of finished batch runs in stage 2 are temporarily stored in tanks of stage 3 for checking of the product quality. The tanks with a relatively large size are allowed to store several batches of products of the same grade at the same time. One example showing the changing volume of products in one of tanks is presented in Fig. 2; fixed-size batches of products are transferred into the tank sequentially with an increasing volume. After the n batches of products are transferred to the tank at time t_{2n}, the products of the same grade are mixed and sampled for lab analysis. When the product has been proven to meet the quality requirement at time step t_{2n+1}, it is further transferred and leaves the tank empty at time step t_{2n+2} for another round of product quality checks. Process parameters related to Fig. 2 include: the time for transferring one batch of product to the storage tank tr, and the time for the product quality check and further transferring t_{qc}.

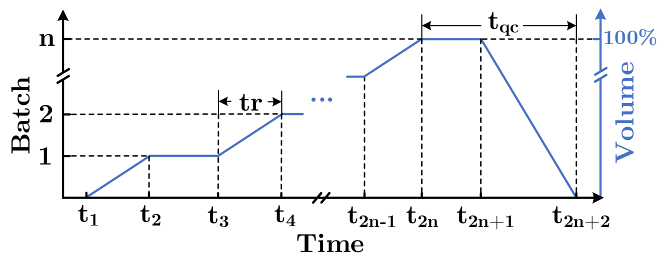

Fig. 2. Time-series change of the product volume in a storage tank; n fixed-size batches fit into the maximum volume of the tank

The storage tanks collect the same grade of products each time with a maximum volume of products to perform the quality check. As a result, a limited number of product grades are allowed to be produced in each time period according to the number of the parallel tanks. Two strategies for the product quality check are considered in the case study: the first one always fills up each tank with the same grade of products before the check; the other strategy allows the quality check to be performed when the tank is not full. The first strategy saves times of product quality checks but adds restrictions to the recipe sequence of batch runs in the other stages, while the second strategy presents the reverse. Both strategies of product quality checks are considered in this work.

3. NOMENCLATURE

The indices, sets, parameters and variables in the proposed scheduling formulation are summarized as follows.

Indices

i, i'	Batch order
r, r'	Batch recipe
j	Unit
l	Stage; l_s denotes the stage of the storage tanks
g, g'	Group of batches stored in a certain tank together for a product quality check

Sets

I	Batch orders
R	Batch recipes
G_r	Groups of batches using recipe r
J	Units
L	Stages
L_p	Subset of production stages
J_l	Subset of units in stage l
J_s	Storage tanks

Parameters

tp_{ij}	Fixed processing time of order i at unit j
ts_j	Start time when unit j becomes available
td_i	Delivery time for order i
tr_{il}	Time for material transfer of order i in stage l to the next stage
t_{qc}	Time for product quality check of a product group and further transferring in stage L_s

Continuous variables

Ts_{il}	Start time of order i in stage l
Te_{il}	End time of order i in stage l
Tp_{il}	Processing time of order i in stage l
Tw_{il}	Waiting (idle) time of order i in stage l
Ts^G_{rg}	Start time of group (r,g) in storage tanks
Te^G_{rg}	End time of group (r,g) in storage tanks
Tw^G_{rg}	Waiting (idle) time of group (r,g) in storage tanks
SG_{rgj}	Size of group (r,g) in storage tank j
MS	Makespan
TK_i	Slacking variable allowing order i delivered before or after td_i
TK^E_i	Earliness variable for delivering order i before td_i
TK^T_i	Tardiness variable for delivering order i after td_i

Binary variables

Y_{ij}	Assignment decision of order i to unit j
Y^S_{irg}	Assignment decision of order i to group (r,g)
$X_{ii'l}$	Sequencing decision for order i preceding order i' at a certain unit of stage l
$X^G_{rgr'g'}$	Sequencing decision for group (r,g) preceding group (r',g') in storage tanks

4. BATCH SCHEDULING MODEL

In this section, scheduling formulations that model the aforementioned case study are proposed. The scheduling problem includes short-term batch scheduling in production units and storage tanks where the product quality checks are performed. Two types of strategies for the checks are considered as two scenarios modeling the interaction between production units and the storage tanks.

4.1 Batch orders in storage tanks

A new representation of modeling batch orders in storage tanks is introduced using a grouping concept. In the short-term scheduling problem, the product orders for various product grades are provided from the supply chain planning as the target of production. The number and sizes of batches are presented according to the orders using indexes $i \in I$ and recipes $r \in R$. Products of batch orders that use the same recipe are transferred to the same storage tank for product quality checks, and the concept of groups is introduced to represent the union of the orders in each individual check. Since groups are recipe-specific, recipe r and group g are combined together to denote individual groups in storage tanks.

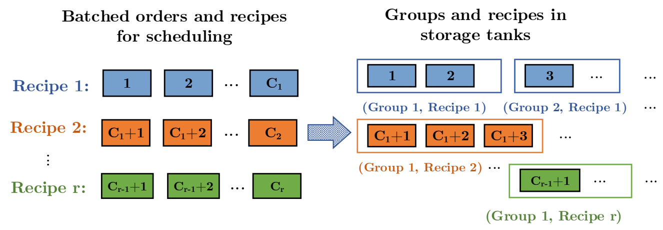

Fig. 3. Indexes of recipes, orders, groups

Figure 3 shows the indexes of batch orders, recipes and groups and the connections among them. An ordered set using recipe r is presented using a specific color in Fig. 3, where C_r denotes the order with the largest index in the order set. A group consists of orders that use the same recipe and have sequential increasing indexes; the indexes (r, g) indicate the tags of groups. Orders using the same recipe are identical with respect to batch operations and production performance. Keeping sequentially increasing indexes of orders in groups prevents other equivalent sequences thereby cutting repeated feasible solutions and improving the computational efficiency.

4.2 Assignment of orders to groups

Two scenarios are considered for grouping batch orders in the storage tanks. The first scenario (SC1) accounts for performing product quality checks in storage tanks only when they are full. The second scenario (SC2) considers product quality checks to be performed flexibly for tanks with any amount of batches inside them and therefore, do not have to wait to be completely filled. Therefore, the size of the groups in SC1 are fixed so that batches are assigned to groups automatically; the size of groups in SC2 is however adjustable, and the corresponding scheduling includes decisions for assigning orders to groups. To assign orders to groups, one of the key variables is Y^S_{irg} which represents assigning order i to group (r, g) when it is true.

Fixed groups (SC1) The size of groups in SC1 is fixed and recipe-specific, which is denoted as SG_r. The assignment of orders to groups is fixed according to SG_r. In this case, the number of batches using recipe r N_r is a multiple of SG_r, where G^{max}_r denotes the number of groups of recipe r which equals N_r/SG_r. Two binary parameters Y^{SF}_{irg} and Y^{SL}_{irg} are introduced to show the fixed assignment of order i in the first place or the last place of group (r, g) as Eqs. (1) and (2) illustrate. Similarly, Y^S_{irg} is known in this scenario as Eq. (3) presents. Furthermore, the predefined groups are assigned to one of the storage tanks as Eq. (4) shows, where Y^G_{rgj} denotes assigning group (r, g) to storage tank j when it is true.

$$Y^{SF}_{irg} = 1, \ \forall\ g \in G_r,\ i \in I_r : i = (g-1) \cdot SG_r + C_{r-1} + 1 \qquad (1)$$

$$Y^{SL}_{irg} = 1, \ \forall\ g \in G_r,\ i \in I_r : i = g \cdot SG_r + C_{r-1} \qquad (2)$$

$$Y^{S}_{irg} = 1, \ \forall\ g \in G_r,\ i \in I_r : (g-1) \cdot SG_r + C_{r-1} + 1 \leqslant i, \quad i \leqslant g \cdot SG_r + C_{r-1} \qquad (3)$$

$$\sum_{j \in J_S} Y^G_{rgj} = 1, \; \forall \, r \in R, \; g \in G_r \tag{4}$$

Adjustable groups (SC2) In SC2 the size of groups is adjustable with a minimum size $SG_{r,min}$ and maximum size $SG_{r,max}$. As a result, the number of groups for each recipe is not fixed represented as $G_r = \{1,2,3, ..., G^{max}_r\}$, in which G^{max}_r is determined by the minimum size of group and number of orders and equals $N_r/SG_{r,min}$. SG_{rgj} represents the size of group $(r, \; g)$ in storage tank j. Equation (5) shows the constraint on the size of the groups, while Eqs. (6) and (7) are the constraints for assigning orders to groups. Equation (8) combined with Eq. (5) defines the assignment of groups to storage tanks. Note that some of the predefined groups may not be assigned any orders thereby leaving them empty. Equation (8) allows orders to fill in groups with smaller indexes and therefore reduces equivalent feasible solutions due to potential symmetry in the set of groups.

$$SG_{r,min} \cdot Y^G_{rgj} \leqslant SG_{rgj} \leqslant SG_{r,max} \cdot Y^G_{rgj}, \quad \forall \, r \in R, \; g \in G_r, \; j \in J_s \tag{5}$$

$$\sum_{j \in J_s} SG_{rgj} = \sum_{i \in I_r} Y^S_{irg}, \qquad \forall \, r \in R, \; g \in G_r \tag{6}$$

$$\sum_{g \in G_r} Y^S_{irg} = 1, \qquad \forall \, r \in R, \; i \in I_r \tag{7}$$

$$\sum_{j \in J_S} Y^G_{rg'j} \leqslant \sum_{j \in J_S} Y^G_{rgj} \leqslant 1, \forall r \in R, g, g' \in G_r : g < g' \tag{8}$$

4.3 Sequencing and timing of batch orders

To schedule the production stages, batch orders are assigned and sequenced at the units of each stage. Binary Y_{ij} denotes assignment of order i to unit j when it is true, and Eq. (9) shows the constraint that allows order i only to be assigned in one of the units in each stage. Using the general precedence concept, the sequencing of batch orders is formulated as a GDP-based logic constraint in Eq. (10).

$$\sum_{j \in J_l} Y_{ij} = 1, \qquad \forall \, i \in I, \; l \in L_p \tag{9}$$

$$\begin{bmatrix} X_{i'il} \wedge Y_{ij} \wedge Y_{i'j} \\ Te_{i'l} + Tw_{i'l} + \\ tr_{i'l} \leqslant Ts_{il} \end{bmatrix} \underline{\bigvee} \begin{bmatrix} \neg X_{i'il} \wedge Y_{ij} \wedge Y_{i'j} \\ Te_{il} + Tw_{il} + \\ tr_{il} \leqslant Ts_{i'l} \end{bmatrix}, \quad \forall \, i, i' \in I : i' < i, \; j \in J_l, \; l \in L_p \tag{10}$$

where, the logic expression $X_{i'il} \wedge Y_{ij} \wedge Y_{i'j} = 1$ indicates order i precedes order i' at unit j of stage l, while $\neg X_{i'il} \wedge Y_{ij} \wedge Y_{i'j} = 1$ indicates order i' precedes order i at unit j of stage l; the corresponding big-M constraints are presented in Eq. (11). Te_{il} is the end time of order i at one of the units in stage l and is computed from the corresponding start times as Eq. (12) shows. The start time Ts_{il} is always after the time when the corresponding unit becomes available as Eq. (13) shows. Tp_{il} in Eq. (12) is the total processing time of order i in stage l and depends on the sequencing and assignment binary variables such as $Tp_{il} = \sum_{j \in J_l} tp_{ij} \cdot Y_{ij}$, where tp_{ij} is the fixed processing time of order i at unit j; material transfer from upstream stage takes up the unit, and therefore $tr_{i(l-1)}$ is included to the processing of order i in stage l; Equation (14) presents the timing of orders being processed in neighboring stages; Tw_{il} denotes the waiting time after processing order i in stage l allowing storage of order i in the current unit before processing it in the next stage.

$$\begin{cases} Te_{i'l} + Tw_{i'l} + tr_{i'l} \leqslant Ts_{il} + \\ \qquad M \cdot (3 - X_{i'il} - Y_{ij} - Y_{i'j}), \\ Te_{il} + Tw_{il} + tr_{il} \leqslant Ts_{i'l} + \\ \qquad M \cdot (2 + X_{i'il} - Y_{ij} - Y_{i'j}), \end{cases} \quad \forall \, i, i' \in I : i' < i, \; j \in J_l, \; l \in L_p \tag{11}$$

$$Te_{il} = Ts_{il} + tr_{i(l-1)} + Tp_{il}, \qquad \forall \, i \in I, \; l \in L_p \tag{12}$$

$$Ts_{il} \geqslant \sum_{j \in J_l} ts_j \cdot Y_{ij}, \qquad \forall \, i \in I, \; l \in L_p \tag{13}$$

$$Ts_{i(l+1)} = Te_{il} + Tw_{il}, \quad \forall \, i \in I, \; l \in L_p : l < |L_p| \tag{14}$$

4.4 Sequencing and timing of groups

Batch orders are assigned and sequenced in the units of the production stages, and groups that consist of batches of the same product grade are assigned and sequenced in storage tanks as the example shows in Fig. 4 (b) instead of the overlapping one in Fig. 4 (a). Like the sequencing of orders, general precedence models can also handle sequencing of groups in storage tanks, which links timing of groups and orders together. The constraints for two scenarios are described separately.

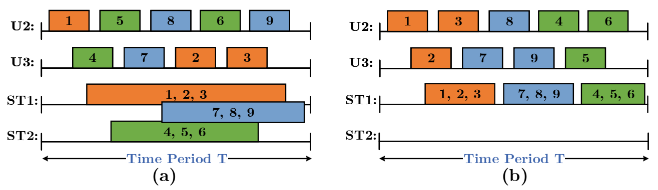

Fig. 4. Sequencing of orders and groups with colors denoting the recipes: (a) overlapping groups in storage tanks ST1 and ST2; (b) no overlapping groups

Fixed groups (SC1) Since binaries Y^S_{irg}, Y^{SF}_{irg} and Y^{SL}_{irg} are known, Eqs. (15) and (16) indicate the exact start and end time of groups that are coupled with the timing of the corresponding orders. The timing and sequencing between two different groups are modeled using general precedence, and the corresponding GDP-based constraints are presented in Eq. (17). The logic expression $X^G_{r'g'rg} \wedge Y^G_{r'g'j} \wedge Y^G_{rgj} = 1$ denotes that group (r', g') precedes group (r, g) in storage tank j when it is true, which is followed with the corresponding disjunction, and (r', g') and (r, g) are differentiated from each other using a index term $100 * r + g$. Tw_{rg} represents the waiting/idling time of group (r, g) between filling the last order of the group and performing the product quality check. t_{qc} is the time cost of product quality check and transferring from storage tanks to further delivery. The logic constraints in Eq. (17) are further reformulated as big-M reformulated linear inequalities as Eq. (18) shows.

$$Te^G_{rg} = \sum\nolimits_{i \in I_r} (Te_{il} + Tw_{il} + tr_{il}) \cdot Y^{SL}_{irg}, \; \forall l \in L_s, \; r \in R, \; g \in G_r \tag{15}$$

$$Ts^G_{rg} = \sum\nolimits_{i \in I_r} (Te_{il} + Tw_{il}) \cdot Y^{SF}_{irg}, \; \forall l \in L_s, \; r \in R, \; g \in G_r \tag{16}$$

$$\begin{bmatrix} X^G_{r'g'rg} \wedge Y^G_{r'g'j} \wedge Y^G_{rgj} \\ Te^G_{r'g'} + Tw_{r'g'} + \\ t_{qc} \leqslant Ts^G_{rg} \end{bmatrix} \underline{\bigvee} \begin{bmatrix} \neg X^G_{r'g'rg} \wedge Y^G_{r'g'j} \wedge Y^G_{rgj} \\ Te^G_{rg} + Tw_{rg} + \\ t_{qc} \leqslant Ts^G_{r'g'} \end{bmatrix}, \quad \forall \, r, r' \in R, j \in J_S, g \in G_r, g' \in G_{r'} : (r, g) < (r', g') \tag{17}$$

$$\begin{cases} Te^G_{r'g'} + Tw_{r'g'} + t_{qc} \leqslant Ts^G_{rg} + \\ \qquad M \cdot (3 - X^G_{r'g'rg} - Y^G_{r'g'j} - Y^G_{rgj}), \\ Te^G_{rg} + Tw_{rg} + t_{qc} \leqslant Ts^G_{r'g'} + \\ \qquad M \cdot (2 + X^G_{r'g'rg} - Y^G_{r'g'j} - Y^G_{rgj}), \end{cases}$$
$$\forall\, r, r' \in R, j \in J_S, g \in G_r, g' \in G_{r'} : (r,g) < (r',g') \quad (18)$$

Adjustable groups (SC2) In SC2 timing between orders and groups are coupled with binary variables Y^S_{irg} as Eqs. (19) and (20) show, that is, the start and end times of a group depend on the end time of batch orders that are assigned to the group. Because of the bilinear terms in Eqs. (19) and (20), a GDP-based constraint in Eq. (21) is introduced and reformulated as big-M linear inequalities in Eq. (22). The constraints on timing and sequencing of two different groups remain the same as SC1 (see Eq. (17)).

$$Te^G_{rg} \geqslant (Te_{il} + Tw_{il} + tr_{il}) \cdot Y^S_{irg},$$
$$\forall\, i \in I_r,\ l = l_s,\ r \in R,\ g \in G_r \quad (19)$$
$$Ts^G_{rg} \leqslant (Te_{il} + Tw_{il}) \cdot Y^S_{irg},$$
$$\forall\, i \in I_r,\ l = l_s,\ r \in R,\ g \in G_r \quad (20)$$

$$\underset{g \in G_r}{\veebar} \begin{bmatrix} Y^S_{irg} \\ Te_{il} + Tw_{il} + tr_{il} \leqslant Te^G_{rg} \\ Ts^G_{rg} \leqslant Te_{il} + Tw_{il} \end{bmatrix},\ \forall\, i \in I_r,\ l = l_s,\ r \in R, \quad (21)$$

$$\begin{cases} Te_{il} + Tw_{il} + tr_{il} \leqslant Te^G_{rg} + M \cdot (1 - Y^S_{irg}), \\ Ts^G_{rg} \leqslant Te_{il} + Tw_{il} + M \cdot (1 - Y^S_{irg}), \end{cases}$$
$$\forall\, i \in I_r,\ l = l_s,\ r \in R,\ g \in G_r \quad (22)$$

4.5 Shared pipes for material transfer

In two neighbouring stages, units share the same transfer pipes resulting in restrictions on material transfers for ongoing batch runs. The sets of units in stage l that only use and share the same pipe k are denoted as J^{tr}_k, $k \in K_l$. The material transfer using pipe k becomes unavailable once the pipe is already occupied by another unit in set J^{tr}_k. The constraint that prevents overlapping of two orders in the same shared transfer pipe is modeled using general precedence, and the GDP-based constraint in Eq. (23) presents disjunctions with binary variables $X_{i'il}$ and Y_{ij}. In the disjunctions, any two orders are assigned to the units of set J^{tr}_k, and the material transfers of both have to be sequenced one after another.

$$\begin{bmatrix} X_{i'il} \wedge (\underset{j \in J^{tr}_k}{\veebar} Y_{ij}) \wedge (\underset{j \in J^{tr}_k}{\veebar} Y_{i'j}) \\ Te_{i'l} + Tw_{i'l} + tr_{i'l} \leqslant Te_{il} + Tw_{il} \end{bmatrix} \veebar \begin{bmatrix} \neg X_{i'il} \wedge (\underset{j \in J^{tr}_k}{\veebar} Y_{ij}) \wedge (\underset{j \in J^{tr}_k}{\veebar} Y_{i'j}) \\ Te_{il} + Tw_{il} + tr_{il} \leqslant Te_{i'l} + Tw_{i'l} \end{bmatrix},$$
$$\forall\, i, i' \in I : i' < i,\ k \in K_l,\ l \in L \quad (23)$$

In the case study example, the two units in stage 2 share the only pipe ($|K_l| = 1$) leading to the constraint in Eq. (24) that is simplified from Eq. (23). The corresponding big-M reformulated constraint is presented in Eq. (25).

$$\begin{bmatrix} X_{i'il} \\ Te_{i'l} + tr_{i'l} + Tw_{i'l} \leqslant \\ Te_{il} + Tw_{il} \end{bmatrix} \veebar \begin{bmatrix} \neg X_{i'il} \\ Te_{il} + tr_{il} + Tw_{il} \leqslant \\ Te_{i'l} + Tw_{i'l} \end{bmatrix},$$
$$\forall\, i, i' \in I : i' < i,\ l \in L \quad (24)$$

$$\begin{cases} Te_{i'l} + tr_{i'l} + Tw_{i'l} \leqslant Te_{il} + Tw_{il} + M \cdot (1 - X_{i'il}), \\ Te_{il} + tr_{il} + Tw_{il} \leqslant Te_{i'l} + Tw_{i'l} + M \cdot X_{i'il}, \end{cases}$$
$$\forall\, i, i' \in I : i' < i,\ l \in L \quad (25)$$

4.6 Objective function

The batch scheduling model can be formulated with different objective functions. one of the objective functions is to minimize makespan MK, and MK is defined in Eq. (26).

$$MS \geqslant Te^G_{rg},\ \forall\, r \in R,\ g \in G_r \quad (26)$$

Just-in-time optimization considers the delivery time for each order and schedules the production in a way that minimizes lead or lag time of the batches. The delivery time constraint is presented in Eq. (27).

$$\begin{cases} TK_i \leqslant td_i - Te^G_{rg} + M \cdot (1 - Y^S_{irg}), \\ TK_i \geqslant td_i - Te^G_{rg} - M \cdot (1 - Y^S_{irg}), \end{cases}$$
$$\forall\, r \in R,\ i \in I_r,\ g \in G_r \quad (27)$$

where, td_i is the delivery due time of order i; Tk_i is a slack variable to allow order i be delivered before or after td_i. Minimization of tardiness is presend in Eq. (28):

$$\min \textstyle\sum_{i \in I} TK^T_i \quad (28)$$

where, TK^T_i is non-negative,

$$TK^T_i \geqslant -TK_i,\ \forall\, i \in I$$

Minimization of earliness is defined in Eq. (29):

$$\min \textstyle\sum_{i \in I} TK^E_i \quad (29)$$

where,

$$TK^E_i \geqslant TK_i,\ \forall\, i \in I;\ \ TK_i \geqslant 0,\ \forall\, i \in I$$

5. RESULTS

In this section, the proposed scheduling formulations are tested using problem instances based on the case study. In the problem instances, the multiproduct batch process has three stages as Fig. 1 shows, where stage 3 consists of two storage tanks denoted as units $ST1$ and $ST2$; each tank can store a maximum of three batches of products; as a result, the fixed group size in scenario SC1 is given as $SG_r = 3$, while in scenario SC2 the range of group size is defined as $SG_{r,max} = 3$ and $SG_{r,min} = 1$; units $U2$ and $U3$ use the same pipe for product transfer to storage tanks; the numbers of recipes and orders are adjustable to test the scalability of the formulation. The scheduling models for the mentioned two scenarios in stage 3 were coded in software GAMS 28.2 with the aforementioned objective functions. The resulting MILPs are solved on a Windows 10 computer with an Intel i5 (2.4Ghz and two cores) processor and 16 GB of RAM using CPLEX 12.9 with four threads.

The names of problem instances MS1 and MS2 denote that the objective function of this problem is minimizing MS. Problem MS-D1 and MS-D2 refer to scheduling problems with the goal of minimizing MS with the due time constraints for product delivery. Given due time for the delivery of batch products the scheduling problems considering a minimization of tardiness are denoted as TAR1 and TAR2, while the ones with minimum earliness are EAR1 and EAR2. The problem sizes of the aforementioned

Table 1. Computational results for batch scheduling with two scenarios

Prob.	$\|I\|$	$\|R\|$	CPU time (sec)		Optimal Obj.	
			SC1	SC2	SC1	SC2
MS1	12 (6,3,3)	3	65.42	97.28	3782	3777
MS2	15 (6,6,3)	3	1357.62	2617.28	4574	4569
MS-D1	12 (6,3,3)	3	0.28	23.33	3805	3797
MS-D2	15 (6,6,3)	3	0.28	178.86	4600	4583
TAR1	12 (6,3,3)	3	2.75	9.14	3553	540
TAR2	15 (6,6,3)	3	9.95	60.7	3971	575
EAR1	12 (6,3,3)	3	0.13	2.33	3129	812
EAR2	15 (6,6,3)	3	1.64	130.72	4248	1509

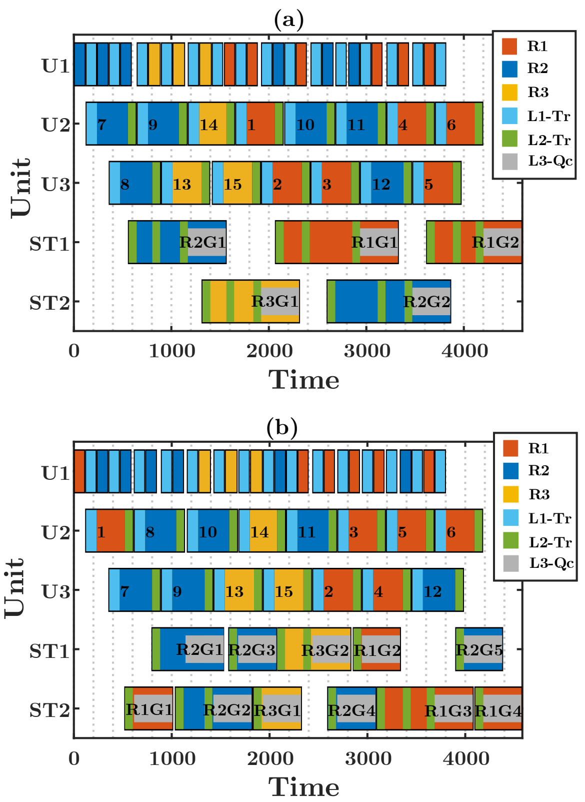

Fig. 5. Gantt charts for MS-D2: (a) Scenario SC1; (b) Scenario SC2

instances are illustrated in Table 1; $|R|$ represents the number of recipes used in the short-term scheduling, and $|I|$ denotes the total number of batch orders with the number of batches of each recipe added in parentheses. Comparing the results in Table 1 between scenarios SC1 and SC2, the models for SC2 present better solutions (smaller objective values) with respect to different objective functions, while the computational cost increases faster as the problem size grows. This is because the batch sequences are much restricted due to fixed groups in the models of SC1, and smaller feasible solution sets in SC1 relatively reduce the computational cost in comparison with SC2. By adding the timing constraints for product delivery on problems MS1 and MS2, the new problem instances MS-D1 and MS-D2 are solved within a much shorter time period. The Gantt charts of the solutions for problem instance MS-D2 are presented in Fig. 5. The solution for scenario SC2 in Fig. 5 (b) performs 11 product quality checks, while the one in Fig. 5 (a) performs only five. To meet the just-in-time requirements, problems TAR1, TAR2, EAR1 and EAR2 are taken as examples for illustrating minimum tardiness or minimum earliness. The corresponding solutions in Table 1 indicate efficacy of the model using SC2. The flexibility of the product storage and quality check in SC2 allow more groups of batch orders to meet the requirements of the due time for the product deliveries. A trade-off exists between meeting the just-in-time requirements (SC2) and performing less checks during production (SC1).

6. CONCLUSIONS

This paper proposed short-term scheduling formulations that consider the storage and material transfer features of the multiproduct batch case study. A continuous-time MILP formulation is applied to model two scenarios of the considered features. The computational results illustrate the efficiency of the scheduling formulations of the two scenarios. In the problems that minimize makespan the solutions using SC2 save less than ten minutes production time but add more than four quality checks comparing to the ones using SC1. However, SC2 performs much better than SC1 for the problems with just-in-time requirements.

REFERENCES

Aguirre, A.M. and Papageorgiou, L.G. (2018). Medium-term optimization-based approach for the integration of production planning, scheduling and maintenance. *Computers & Chemical Engineering*, 116, 191–211.

Biondi, M., Sand, G., and Harjunkoski, I. (2017). Optimization of multipurpose process plant operations: A multi-time-scale maintenance and production scheduling approach. *Computers & Chemical Engineering*, 99, 325–339.

Dalle Ave, G., Hernandez, J., Harjunkoski, I., Onofri, L., and Engell, S. (2019). Demand side management scheduling formulation for a steel plant considering electrode degradation. *IFAC-PapersOnLine*, 52(1), 691–696.

Gupta, S. and Karimi, I. (2003). An improved milp formulation for scheduling multiproduct, multistage batch plants. *Industrial & engineering chemistry research*, 42(11), 2365–2380.

Kilic, O.A., van Donk, D.P., and Wijngaard, J. (2011). A discrete time formulation for batch processes with storage capacity and storage time limitations. *Computers & Chemical Engineering*, 35(4), 622–629.

Kondili, E., Pantelides, C., and Sargent, R. (1993). A general algorithm for short-term scheduling of batch operations—i. milp formulation. *Computers & Chemical Engineering*, 17(2), 211–227.

Méndez, C.A., Cerdá, J., Grossmann, I.E., Harjunkoski, I., and Fahl, M. (2006). State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Computers & Chemical Engineering*, 30(6-7), 913–946.

Méndez, C.A. and Cerdá, J. (2003). An milp continuous-time framework for short-term scheduling of multipurpose batch processes under different operation strategies. *Optimization and Engineering*, 4(1-2), 7–22.

Sundaramoorthy, A. and Maravelias, C.T. (2008). Modeling of storage in batching and scheduling of multistage processes. *Industrial & Engineering Chemistry Research*, 47(17), 6648–6660.

Wu, O., Dalle Ave, G., Harjunkoski, I., Bouaswaig, A.E., Schneider, S.M., Roth, M., and Imsland, L. (2020). Optimal production and maintenance scheduling for a multiproduct batch plant considering degradation. *Computers & Chemical Engineering*, 135.

Wu, O., Dalle Ave, G., Harjunkoski, I., Imsland, L., Schneider, S.M., Bouaswaig, A.E., and Roth, M. (2019). Short-term scheduling of a multipurpose batch plant considering degradation effects. In *Computer Aided Chemical Engineering*, volume 46, 1213–1218. Elsevier.